INVESTIGATION INTO THE SYSTEM OF FUEL MANAGEMENT IN THE JOHANNESBURG CITY ENGINEERS DEPARTMENT AND THE DESIGN OF A MICROPROCESSOR BASED SYSTEM TO ASSIST IN THE CONTROL THEREOF.

DALE STEVEN DRUCKMAN

A dissertation submitted to the Faculty of Engineering, University of the Witwatersrand, for the degree of Master of Science in Engineering.

JOHANNESBURG 1984.

ABSTRACT

This dissertation describes an investigation into the method of recording and control of the fuel utilised by individual vehicles in the City Engineers Department of Johannesburg and the design of a suite of programs to provide statistical information based on this data to enable management to make decisions and take meaningful action towards better control of fuel usage.

The findings of the investigation are given together with recommendations to improve both the security of fuel issues and the accuracy of data recording. Some alternative data capture methodologies are discussed both as long term and short term solutions.

The suite of programs has been designed on a microprocessor, ease of operation being the major concern. The intention is to implement the system in each of the branches of the Department. A description of the design and operation of the system is given together with a worked example of the input screens and output graphs for the various techniques.

Finally, supporting theory, program listings and other relevant material, are given in the Appendices.

DECLARATION

I DECLARE THAT THIS DISSERTATION IS MY OWN, UNAIDED WORK. IT IS BEING SUBMITTED FOR THE DEGREE OF MASTER OF SCIENCE IN ENGINEERING IN THE UNIVERSITY OF THE WITWATERSRAND, JOHANNESBURG. IT HAS NOT BEEN SUBMITTED BEFORE FOR ANY DEGREE OR EXAMINATION IN ANY OTHER UNIVERSITY.

DALE STEVEN DRUCKMAN

31ST DAY OF AUGUST 1984.

DEDICATION

I should like to thank the Statistical Services Branch of the Johannesburg City Engineers Department for the opportunity which occasioned the writing of this paper, and the staff of that department for the assistance given to me in familiarising myself with the operation of the system. My thanks also to my supervisor, Prof. J Bicheno, without whose guidance and assistance this project would not have been possible. Finally to my girlfriend, Sue Rosenberg, for her constant support throughout the project and to my parents, Phyllis Saunders and Sydney Druckman, for their guidance through the years.

CONTENTS

VOLUME I

1. INTRODUCTION

1.1 Background

Over the past few years, it has been increasingly evident that the gravity of the energy shortage both in this country and worldwide is not likely to diminish. This has been accompanied by a growing awareness of the need to control fuel consumption both from the point of view of conservation and that of the high cost of fuel. In many spheres of commerce and industry, companies have begun considering their fuel bills far more seriously than they did in the past, and have begun taking active steps to minimise these costs.

The Johannesburg City Engineers Department have always, due to the size of their investment in their vehicle fleet, been aware of fuel costs and in consequence, have established an extensive system for control and management of fuel usage. However, this system has been in existence for some years and after due consideration, it was decided by management that an in depth investigation was needed to point out some of the loopholes which inevitably occur, over a period of time, in a system which has mutated to meet the needs of a changing environment. It was furthermore felt that a method had to be found to pinpoint with a far greater degree of accuracy and timeousness, instances in which variations in fuel consumption indicated a problem area. This brief was consequently given to the writer.

1.2 Area of Investigation

One of the areas in which the concept of adjusting a process to be within limits by means of monitoring its output has found great application, is in the field of Quality Control. It is therefore somewhat surprising that so little use is made of the application of Statistical Quality Control techniques in the monitoring of fuel usage in large vehicle fleets. This is not to say that there is a lack of vehicle/fleet management systems. However, in almost all cases, no more than a simple comparison between actual and budget usages, or of specific consumption to average group consumption is made and various reports are generated, based on this.

Prior to commencing the investigation an in depth literature survey was undertaken. This survey confirms the above observations. It furthermore shows very little available information on existing manual fuel data collection systems for batch entry related environments. An overwhelming quantity of information is, however, available in the literature on the subject of fuel economy in areas of:

- exhaust emission
- control of vehicle mass
- electronic fuel regulation
- driving style
- route layout/planning

and other technical aspects of engine efficiency. Most of these are not directly relevant to the present investigation, although some useful ideas have been gleaned from a perusal thereof.

The areas which have been investigated in some depth to provide the necessary background for the project are described in the following section.

2. LITERATURE SURVEY

2.1 Computer Aided Fuel Monitoring In Fleet Management

In recent years, with increasing fuel costs and scarcity, the monitoring and control of fuel consumption has become a major factor in overall cost control within most organisations. The major problem has been the prohibitive cost of collecting and co-ordinating the facts and figures required to manage a large fleet, especially in view of the fact that the accuracy required is of great importance.

2.1.1 The explosion in the world of credit cards led naturally to their use in the purchase of fuel and since most credit card systems are computerised, the step to computerised fuel management was a short one. The basic philosophy behind these systems is that each vehicle in the fleet is issued with a "credit card" and a set of credit card vouchers. Whenever a purchase or repair is carried out on that vehicle, the driver is obliged to complete a voucher, which is then submitted by the dealer to the credit card organisation. Payment is made in the same manner as any other credit card operation (Systems - June 1983).

It is interesting to note that to date South Africa appears to be the leader in the development of such systems, probably due to the combination of the numerous large company owned fleets of vehicles and the high fuel prices found in South Africa. Although this combination is not unique to South Africa, this country is probably the most economically advanced of those in which this combination does occur.

At this time the two major systems in existence are the Barclays Auto System developed in 1978 and the Stannic Fleet Management System released in 1980, although a new system recently introduced by Volkskas appears to be gaining popularity. Although there are technical differences, the basic output of the three systems is the same. Every month the customer receives a series of reports giving the following information.

A detailed report of costs for each vehicle for the month, including fill-ups, services and repairs, together with a comparison of the averaged 12 month totals to date.

A variance report which shows any deviations for that vehicle from previously selected limits as well as a comparison of each vehicle's performance with the district and national averages for that vehicle type.

Other reports such as total fleet statistics, showing totals for the entire fleet; vehicle replacement reports and preventitive maintenance reports showing which vehicles are overdue for maintenance operations.

The major disadvantage is the setting up cost, which for any medium sized company prohibitive. Consequently many such companies contract into such a scheme, which might be run by one of the major finance houses, thus avoiding the overheads while still benefiting from the advantages.

With the advent of the mini and micro computer, a far greater awareness of the capability as well as the reduced cost of such machines has persuaded many companies to make more use of smaller computers in the control of vehicle fleets. One such system utilises a micro computer which is installed in the vehicle and records engine revs, distance travelled, fuel used and other trip related data (see Appendix D). At the end of the day the on-board computer is plugged into a desk top computer which extracts all relevant information and then produces such reports as a vehicle performance report, vehicle service schedule, fuel usage analysis and trip reports.

Another such system has been developed by Doctor (1983) in which he utilises a micro computer to produce fleet management information for a medium sized fleet of cleansing vehicles. Facilities included in the system are reports on fuel usage, repair costs, preventive maintenance schedules and vehicle replacement analysis.

A number of other commercially available systems do exist in South Africa.

[illegible]

- Transport Military Services
- I.C.L. Fleetmaster
- Mohawk Data Services Trans-Pac
- Systime Fleetman

These systems run on small to medium sized computers and provide information on vehicle history, preventive maintenance, current costs and consumption data, although storage space is somewhat restricted.

The problems of accurate data collection and reasonable statistical analysis are, however, not addressed by any of these systems.

2.1.2 Another solution which has been shown to work well in Las Vegas, Nevada, at Silver State Disposal Incorporated (World Wastes - July 1983) utilises the principle of connecting each pump through a control board to the main computer.

The driver inserts a precoded card into the relevant slot and using a pin number, activates a transaction request. The computer will then request all relevant data such as registration number, odometer reading, pump number and drivers number, and validates this information before activating the requested pump.

[illegible]

(a) Does the requested pump contain the correct fuel for the specified vehicle.
(b) Is this driver licenced to drive this vehicle
(c) Is the card inserted valid and does it match the pin number entered.
(d) Is the odometer reading entered reasonable.
(e) No more than the total capacity of the vehicle will be dispensed.

Security is maintained by invalidating cards of ex employees and old or lost cards. Furthermore any unauthorised attempted entry is logged and reported. Quick activation and deactivation of pumps prevents unauthorised vehicles from pulling in in front of or behind authorised vehicles and obtaining fuel without first going through the card identification process.

2.1.3 The most important breakthrough in our time in control systems in transportation, according to Sweet (1981) is the development of microprocessor-based control systems. He attributes this to such causes as scarcity and cost of fuel and rapid development of low cost microprocessors. He divides these into two types:

(a) Open Loop Schedule - Type Controllers, in which certain engine variables are adjusted as pre-set functions of other engine variables. Control of air-fuel ratio is an example of this type, where these functions are determined by using optimal control theory to minimise fuel consumption over a specified cycle.

(b) Closed-Loop Adaptive Controllers, in which some variable is controlled by adjusting it, based on a detected deviation in some other variable from a desired condition. Modulation of fuel flow often utilises this technique to control air-fuel ratio.

2.1.4 Another new area of fleet management is that of vehicle trip simulation, where a major breakthrough has been achieved by Cummins Diesel (Bolton 1983) in their development of a system to simulate the performance of various combinations of trucks and trailers over specified routes in order to enable them to select the most cost-effective combinations. Although the design of the system was not initially based on fuel cost savings, this has now become one of the major areas of cost reduction.

2.2 Physical Methods of Fuel Conservation

A vast amount of work has been done in improving physical design parameters of vehicles utilising fuel by, amongst others, Bartlett (1980), La Pointe (1973) and Janssen (1978). Some of the more common areas of research are:

(a) Tyre pressure and tyre widths
(b) Rolling resistance (in large vehicles)
(c) Aerodynamic drag (in non-streamlined loads)
(d) Braking losses

Of these, Bartlett (1980) and La Pointe (1973) claim that aerodynamic drag and rolling resistance are the most marked causes of increased consumption, although in smaller vehicles the effects of all of these factors are insignificant.

Another area of investigation has been that of weight reduction, there being a definite relationship between mass moved and energy expended in moving it. Such heavy items as batteries and chain links are being manufactured from light weight materials and in many instances, are undergoing complete redesign.

2.3 Other Factors In Fuel Consumption

2.3.1 Driver Effectiveness

One of the most important factors in the consumption of fuel is the attitude and level of training of the driver. Smiley (1983) quotes a figure of 36% reduction in maintenance and 17% reduction in fuel costs as average for a good safety training program in the United States. Other authors support this view although there appears to be little in the way of statistical proof for such hypotheses. On a test track set up in Berkshire, where drivers were asked to drive normally and then carefully, a difference in fuel consumption of as much as 14% was measured. Furthermore, the difference in fuel consumption between the worst and best drivers was 50%. Ratcliffe (1980) believes that this "lack of caring' in most commercial vehicle drivers is due to a management attitude of keeping maintenance costs to a minimum and maximising profit by high vehicle utilisation to the detriment of relevant safety factors.

2.3.2 Ambient Temperature Conditions

The relationship between fuel consumption and ambient temperature is quantified by Hayden (1979) who shows that fuel consumption of petrol vehicles at -12 degrees C is more than 200% of the consumption at 4 degrees C while diesel vehicles show a slightly smaller increase (180%). With increasing severity of temperature this difference becomes more marked.

An interesting factor is the length of time taken to reach normal operating temperature. Waters (1980) shows that tyres and transmission are the limiting factor in larger vehicles since they take longer to warm up than does the engine. To some extent then, it appears as if this excess consumption is unavoidable.

2.3.3 Diesel vs Petrol - Which to use

Hayden (1979) shows, using a series of weight normalised curves, that diesel vehicles have an advantage in fuel consumption of 57% at 21 degrees C and an even greater advantage (92%) at 12 degrees C. Waters (1980) points out that this should be offset against the higher initial cost of the diesel vehicle as well as the fact that trade-in values are much lower. He also comments that although diesel fuel contains more energy per litre than does petrol, less energy is expended to produce it. In terms of overall energy consumption therefore, the two are approximately on a par. Of course this is relevant only to studies of general energy conservation and not specifically to the attempts to reduce fuel costs by reduced consumption.

2.3.4 Vehicle Design

A great deal of research is being undertaken, especially by the major vehicle manufacturers, in improved design of combustion systems, transmission systems, exhaust and emission systems and other areas. These are not of relevance to this work and are therefore excluded.

2.3.5 Fuel Density, Volatility and Octane Rating

Fuel dispensed into most commercial vehicles is seldom homogeneous, specifically in terms of density. This leads to changes in consumption of fuel. Bascunana and Stahman (1978) claim differences of up to 7% in Michigan, USA, in tests conducted in 1976/77, although other authors do not support this claim, saying that these differences are not measurable.

Fuel volatility and octane rating play a much smaller direct role in fuel consumption. However, volatility greatly effect the warm-up period and thus indirectly, the consumption.

Factors such as fuel viscosity and surface tension are believed to have a minor effect but these are difficult to evaluate and no data is readily available to quantify these effects.

2.3.6 General

In his discussion on fuel conservation policies, Flachsbart (1979) points out the following principles:

(a) The fact that to a large extent vehicles are used as a means for communicating. Substituting more modern methods of communication would therefore certainly save fuel. It appears though that this would only be practical if implemented at community level but costs involved would be prohibitively high.

(b) Other methods of fuel conservation are lane reservation for high-occupancy vehicles and car-pool programs. The former has worked with some success in Los Angeles and with a good deal of success in Sydney, Australia. The latter, however, depends on the goodwill of the community and has the inherent disadvantage that a car left at home might be used for more frequent, less economical trips than it otherwise would have been.

2.4 Statistical Methods In Quality Control

Throughout the literature, a number of statistical techniques emerge as being useful in Quality Control. These are:

- Shewhart (Control) Charts
- Cumulative Sum (Cusum) Charts
- Moving Averages

- Regression Analysis
- Exponential Smoothing

Naturally, the choice of technique depends largely on the purpose for which the information will be used, and the data being used as input. A discussion on each method and its uses follows:

2.4.1 Shewhart (Control) Charts

In 1924, Dr W.A. Shewhart established a graphical method of distinguishing between the random chance factors which affect a process and those which are due to some specific cause. This graphical control chart may then be used to determine whether or not a given process is in control, i.e. within the bounds of a previously specified percentage of probability, the values of the process will fall within certain limits.

Wetherill (1977) points out that this philosophy, although at first appearing to be naive, is borne out by a long history of widespread applications and is shown to be of great value.

Usually a control chart is constructed by choosing a variable which bears some relationship to the quality of the process, and plotting it against time. The mean is calculated using historical data and the current values of the process are compared with this mean. In order to determine the limits outside of which the probability of a chance variation is so small as to be almost negligible, the history of variation of the process is important. It has been shown that, assuming the process is normally distributed, this probability is somewhere around 0,003 (Grant 1980). In fact most often these limits, called control limits or action lines, are chosen to be at a distance of 3 standard deviations (3σ) away from the mean (see Appendix A.1 for discussion of standard deviation). Some probabilities for the normal curve are reproduced from Grant (1980).

Table 2.4.1

Limits	% of Total Area Within Limits
$\bar{x} \pm \sigma$	68,26
$\bar{x} \pm 2\sigma$	95,46
$\bar{x} \pm 3\sigma$	99,73

Furthermore, a non-normal universe shows an apparently normal distribution of sample means (see Appendix A.2 for discussion of Central Limit Theory). Although this distribution is not exactly normal, the deviation is not significant, and in fact it approaches normalcy with increasing sample size.

In order to determine with certainty whether a given process is "in control" it is necessary to compute limits for both variations in the mean ($\bar{x}$) as well as variations in the standard deviation (σ). This latter chart is usually substituted by a chart for the range of the occurrences within a sample (R) since for small sample sizes (less than 25) the estimation of the process standard deviation

(σ) from the range is sufficiently accurate, and it is both easier to calculate and to understand the range chart, than the σ chart.

The control limits for the mean are computed as follows:

$$UCL_{\bar{x}} = \bar{\bar{X}} + A_2\bar{R}$$

$$LCL_{\bar{x}} = \bar{\bar{X}} - A_2\bar{R}$$

Here the factor A_2 is derived from:

$$A_2 = \frac{\psi}{d_2\sqrt{\kappa}}$$

where ψ = $1 - \alpha$ quantile of the standardised normal distribution.

κ = subgroup size

d_2 = expectation of the range of a group of size n taken from a standardised normal distribution

The proof for the factor d_2 is beyond the scope of this paper but may be found in Sarkadi (1974).

The control limits for the range are computed from:

$$UCL_r = D_4\bar{R}$$

$$LCL_r = D_3\bar{R}$$

The factors D_3 and D_4 are derived from:

$$D_4 = \frac{W_4}{d_2}$$

$$D_3 = \frac{W_3}{d_2}$$

Where W_4 and W_3 are the ,00135 and ,99865 quantiles of the distribution of the range of a group of size n from the standardised normal distribution and d_2 = expectation of the same range.

Any occurrence or group of occurrences (see Appendix A.4 for theory of extreme runs) which falls outside of these limits is considered to be outside of the probability of a chance variation and must be investigated for assignable causes. A table of these factors A_2, D_3 and D_4 is shown in Appendix A.5 for 3σ probability limits.

The economic design of control charts forms the basis of another wide area of research. Heikes and Montgommery (1981) discuss the utilisation of optimization techniques to determine the parameters for the control charts to minimise costs. They also point out a method of modelling autocorrelated data with an ARIMA (see Exponential Smoothing) time series model, and then constructing a conventional control chart to monitor the residuals.

Grant (1980) believes that in practice, the difficulties involved in determining accurate estimates of the costs involved, negate any advantage gained in the attempt. Taylor (1965) has shown that any chart using constant size samples at fixed time intervals is not optimal, however, most other authors ignore this theory due to the practical difficulties it would create. From a model developed by Chiu and Wetherill (1974), Montgommery (1980) puts forward certain conclusions in which he shows the relationships between the various cost elements and the different design parameters. His most important conclusion is that the model is sensitive to errors in estimation of magnitude of shift from the in-control state to the out-of-control state and the standard deviation, but is insensitive to the errors in cost estimates. He believes that the existence of so few implemented models is due to the complexity of the mathematical models themselves as well as the difficulties in estimation of the various model parameters.

One further area of control chart techniques which has not been covered is that of multiple assignable causes. Although this is dealt with in the literature, its implementation is complex, is difficult for the uninitiated to understand and is beyond the scope of this work.

2.4.2 Cumulative Sum (Cusum) Charts

This technique has been developed in order to more clearly see small variations in a process mean, which in control charts would be masked by the general variability of the data. The technique uses a cumulative total of the deviations from a reference or mean value in successive data points. Thus for a series of points $x_1, x_2 \ldots x_r$, we have from Murdoch (1979) and Wetherill (1977):

$$S_1 = (x_1 - k)$$

and

$$S_2 = (x_1 - k) + (x_2 - k)$$
$$= S_1 + (x_2 - k)$$

then by extrapolation

$$S_r = \sum_{i=1}^{r} (x_i - k)$$
$$= S_{r-1} + (x_r - k)$$

where S_r = cusum for r occurrences

k = reference value chosen

Now if the mean of all occurences is μ, then:

$$\sum_{i=1}^{r} \frac{x_i}{r} = \mu$$

and

$$\sum_{i=1}^{r} (x_i - k) = \sum_{i=1}^{r} x_i - \sum_{i=1}^{r} k$$
$$= r\mu - rk$$
$$= r(\mu - k)$$

and

$$S_r = S_{r-1} + (x_r - k)$$
$$= r(\mu - k)$$

where r = number of occurrences

Now if plotted against r, the slope of this curve is represented by $(\mu - k)$ and thus the greater the difference of the actual mean from the reference mean, the greater will be the slope of the curve.

[illegible] methods are available to determine [illegible] change in a cusum chart is sufficiently [illegible]ficant to warrant some action.

The V-mask technique consists of superimposing on the chart, a V shaped mask pointing horizontally forward, and situated at a distance d ahead of the latest point with the arms of the mask making an angle σ with the horizontal. The choice of d and σ determine the sensitivity of the mask and use is made of average run lengths to determine these parameters (Wetherill 1977). If any previously plotted points fall outside of the V, the process is assumed to be out of control.

A similar method, the parallel mask technique, utilises a line drawn at horizontal distance d from the latest point at angle σ with the horizontal. A parallel line is drawn through the latest point. Any previously plotted points falling outside of the mask show lack of control.

The decision interval technique, based on the same theoretical principles as the above two techniques (Murdoch 1979). Here a value ,k, is determined and when an observation exceeds this value, a chart is begun. If the chart reverts to zero the process is in control. If, however, the chart now exceeds a value, h, corrective action is required. Again, use is made of a set of ARL curves to determine values of h and k.

All the above methods assume the process to be normally distributed with a scale whose one horizontal unit is equal to two standard deviations on the vertical scale.

These methods are all time consuming and in practice it is often found that speed of interpretation takes precedence over accuracy. Hence it often occurs that the cusum is simply plotted, the slopes are critically evaluated and a decision made on whether corrective action is required or not.

2.4.3 Moving Averages

The objective in the utilisation of this technique, is to eliminate all unwanted fluctuations in data, in order to establish a trend. A base period is chosen which is just long enough to achieve this (the longer the base period the less sensitive is the graph to the data), usually 12 months, and an average is calculated each month using the latest twelve occurrences, i.e. including the newest and dropping off the oldest (Riggs 1982) occurrences.

Two possibilities arise:

(a) The moving average may be plotted as it stands against time and viewed as simply an average which moves forward to always cater for the last k observations. Here the M.A. has the form

$$MA_i = \frac{(x_{i-k+1} + x_{i-k+2} + \ldots x_i)}{k}$$

where k = moving average indicator
x_i = individual observations

Now in order to remove cyclical variations which might occur, each observation may be weighted by a predetermined factor.

$$k(MA_i) = \left[(c_{i-k+1}.x_{i-k+1}) + (c_{i-k+2}.x_{i-k+2}) \_\_\_\_ (c_i.x_i)\right]$$

where $(c_{i-k+1}) + (c_{i-k+2}) + \_\_\_\_ c_i = 1$

(b) A refinement which has proved successful (Grant 1980) utilises the control chart concept of moving average curves. Action lines are placed at the 3 standard deviations limits on both mean and range charts which are calculated in the normal way.

The disadvantage of utilising weighting factors is the difficulty of determining what each individual factor should be. Moving average graphs are often plotted with no weighting factor initially until some cyclical trend becomes evident, at which time the weights are designed using the available data.

2.4.4 Exponential Smoothing

So far we have considered the result or outcome of a given set of circumstances with the intention of making some valuable deductions therefrom. The technique of exponential smoothing uses these historical observations as a method of forecasting what is likely to happen in the future.

Box and Jenkins (1970) have done much work in this field where they propose a general class of models called Autoregressive Integrated Moving Average models (ARIMA). They show that exponential smoothing is a special case of this class.

Here, the new observation is considered to be a function of the previous forecast plus some portion, α, of the difference between the forecast and actual values in the previous period. Thus:

$$\begin{aligned} Fn &= \alpha\ Yn_{-1} + (1-\alpha)\ Fn_{-1} \\ &= Fn_{-1} + \alpha(Yn_{-1} - Fn_{-1}) \end{aligned}$$

where Yn_{-1} = previous actual value
Fn_{-1} = previous forecast value
α = autoregressive factor
Fn = new forecast value

This technique is of use in many situations where some prediction needs to be made of the trend most likely to be followed by a given process.

2.4.5 Regression Analysis

It is often insufficient to characterise a group of data points by a trend line. A very successful method of fitting a line through a series of data points such that the curve connects the mean values of the distribution of each occurrence is known as regression. This greatly facilitates the ability to forecast, at a future time, the dependent variable from the independent one.

Initially it is necessary to determine the form of the curve i.e. linear, quadratic, exponential, etc. Then the unknown coefficients are determined, usually by the method of least squares. Now assuming a linear relationship, the curve is of the form:

$y = a_0 + a_1 x$

and any error in the predicted value is:

$ei = yi - (a_0 + a_1 xi)$

thus sum of squares = $S = ei^2 = \sum (yi - a_0 - a_1 xi)^2$

and by differentiating partially $\frac{\partial s}{\partial a_0}$ & $\frac{\partial s}{\partial a_1}$

and setting equal to zero, S can be minimised and the resulting equations solved for a_0, a_1:

$n a_0 + a_1 \sum xi = \sum yi$

$a_0 \sum xi + a_1 \sum xi^2 = \sum xiyi$

and solving gives

$a_0 = \bar{y} - a_1 \bar{x}$

$a_1 = \frac{\sum (xi-\bar{x})(yi-\bar{y})}{\sum (xi-\bar{x})^2}$

The following conditions are prerequisites in order for these least squares estimates to be the most likely:

(a) The dependent variable, x', must follow a normal distribution for a given yi with mean

$a_0 + a_1 xi$

(b) The variance of the distribution of y is not dependent on the value of x i.e. it is constant for all y values.

These last two statistical forms differ from the first three in that they are used to predict future values of a function based on present and past data, whereas the purpose of the others is to analyse the past and present data in order to evaluate factors relevant to the present.

2.5 Quality Control and Statistical Methods in Fuel Conservation

Surprisingly, there appears to be very little in the literature in the area of statistical quality control procedures utilised for energy or fuel management. A number of minicomputers are, however, available with the capability of producing some of the statistical measures discussed in the previous section (Caporal).

These are complex systems and would require a good deal of "tailoring" to be acceptable to the layman.

One example of the use of such methods for fuel management is discussed by Kreft (1980) who cites some examples of savings made in fuel consumption by some American companies through the use of control charts. A seasonal index is calculated using the ratio of actual to moving average. This index is used to remove seasonal fluctuations after which the normalised data is analysed using conventional control chart techniques.

One of the major difficulties encountered, according to Kreft, is that of persuading the entire organisation to accept the principles behind and goals of the system. The success of the system, he believes, is dictated by the success of overcoming this difficulty.

3. INVESTIGATION

An investigation was carried out into the system of fuel data collection and management at present in existence in the Johannesburg City Engineers Department. This investigation embraced areas such as control of fuel stocks, issue of fuel, method of data collection, input of data into data processing resources and production, distribution and usefulness of management information from data processing resources. A detailed account of the investigation is given below:

3.1 Structure of Department

The City Engineers Department (CED) is structured into a number of major branches. Each of these branches is divided into Depots according to geographic location. It was decided, due to the fact that management of fuel in each of the branches is handled on a similar basis, to confine the study to a specific branch, utilising it as a pilot study which could then be expanded to cover all branches if it proved effective. The branch chosen was the Cleansing Branch, it being the greatest single fuel user and therefore the most likely to benefit from such an investigation.

Some considerable time was spent with each of the four depots in the Cleansing Branch as well as with the Statistical Services and Data Processing Branches in an attempt to understand and document the relevant procedures.

3.2 Issue of Fuel and Data Collection

A system of books of coupons has been established. The face of the coupon shows the literage dispensed, and coupons are printed in 5, 10, 20 and 50 litre types, with petrol and diesel being designated by different colours. These books are issued to each depot as requested. An example is shown in Appendix B.1. At the depot, prenumbered coupons are issued from the book. In the case of diesel, the stub and coupon are filled in by the driver, whereas for petrol, they are filled in by an office clerk. The stub, containing fleet no., fuel quantity, vehicle registration and date remains in the book.

The driver of the vehicle presents the coupon containing branch code, date, fleet number, vehicle registration number and kilometre reading, to the pump attendant (at a pump at which the depot is authorised to fill up) who dispenses the quantity of fuel reflected on the coupon into the vehicle. The pump attendant completes the coupon with the "pump start" and "pump finish" readings, as well as the pump code, and signs and retains the coupon for batching. In the case of smaller machinery, e.g. grass cutters, the attendant dispenses the correct amount into a jerrycan which is transported to a more convenient site and is then used to fill the machines by the machine operators.

Tank dip readings and pump readings are taken and recorded at the beginning and end of each day and these must agree after taking into consideration the quantity of fuel delivered into the tank. An example of this reconciliation statement is shown in Appendix B.2.

Some exceptions do exist as in the case, for example, of vehicles which remain at a tip site. These vehicles are filled from a mobile tanker and coupons are issued from a book held at the tip.

Each month the coupons, together with the reconciliation sheets, are returned to the central co-ordinating office, where the issues on the reconciliation statements are checked against the total of issues reflected on the coupons, and are changed if incorrect. Receipts are verified against the supplier delivery notes which must be attached (this field on the form may also be changed if not in agreement with delivery notes). The reconciliation statements are then sequentially numbered and sent to the Statistical Services Branch, where they are batched and entered into the computer.

3.3 Data Processing and Reporting

At the end of the month, a set of printouts is produced by Statistical Services showing each reconciliation statement by number. The first listing, a pump reconciliation, shows initial pump reading, final pump reading, corrected issues (from issues as discussed above) and any variances. The tank capacity is also shown. The second, a tank reconciliation, shows start and end dip totals, corrected receipts (from delivery notes as discussed above) and any variance.

The reconciliation statements are manually checked to ensure that all pump readings are sequential in any batch submitted, as well as from month to month on the reconciliation listings provided. Any fuel debited but not delivered to the tanks should be obvious from a comparison of these listings and the suppliers debit notes, while any fuel issued for which no coupon is received, will show as a variance between pump readings and corrected issues.

The coupons themselves, after being used to check the reconciliation sheets, are batched together and sent to the Data Processing Branch where they are entered, in batch-entry mode, onto the Councils' Fleetmaster system. As each coupon is entered it is validated against a parameter card containing all relevant information for a given registration number e.g. fleet number, department, branch, etc. before being accepted.

At the end of the month, a voluminous series of management reports is produced, the most important of which are:

3.3.1 Fuel and Oil Consumption

This shows per branch and fleet number, the fuel and oil used in the month, total distance travelled over past twelve months and consumption (L/100km or L/hr), the same figures for the last three months, and fleet averages. Totals per branch are given. This is considered to be a management report only.

3.3.2 Branch Report

This shows per depot and fleet number, the latest kilometre reading, life distance, distances in each of the last three months, litres used this month and consumption (L/100 km or L/hr) this month. Each depot receives the relevant portion of the report each month.

3.3.3 Lists of vehicles in various different sequences; a master list of all information on the parameter cards and a summary showing fuel used per branch for the last 12 running months. This latter report is used by upper management for decision making.

4. FINDINGS

4.1 Fuel Issue and Data Collection

4.1.1 Very often, in practice, the person who dispenses the fuel into the vehicles also takes the pump and tank readings. This could obviously lead to misuse especially in view of the fact that in some instances the book is kept by the pump attendant and is filled in by the driver and pump attendant in conjunction.

It is also theoretically possible, assuming some co-operation between driver and attendant, to dispense a quantity of fuel into the vehicle, different than is indicated on the pump, the only check on this being the consumption statistics produced by the system, which as will be shown later, are inaccurate.

4.1.2 Should the vehicle not take the stipulated quantity, it rests upon the pump attendant to alter the coupon to agree with the quantity actually disbursed, while no record is made in the stub.

4.1.3 Although the coupons are prenumbered, no sequential control is kept since this has proved too difficult in the past. It is therefore possible for coupons to go missing entirely, possibly to be used at another depot, although it is unlikely that this could give rise to any irregularities unless once again the driver and pump attendant co-operate to falsify the registration and fleet details on the coupon.

4.1.4 In some cases, the data kept by the depot on its fuel usage s supplied in summary form at month end by the local pump personnel who summarise this information from the coupons. It often happens, however, that vehicles from another depot or even other branches have filled up at the pumps and these summaries consequently include the amounts on these "foreign" coupons.

In other cases, the data summaries kept by the depot on its fuel consumption are drawn up from the coupon stubs. This is a more accurate indicator of the fuel usage of the depot but has the disadvantage that no overfills will be taken into account.

4.1.5 Although the golden rule appears to be "no coupon, no fuel" it nevertheless happens, due to the fact that some coupons contain no preprinted department code, that a vehicle fills up at a "foreign" depot and may even use one of their coupons. This would obviously invalidate any data taken from that stub, for that depot. It raises the further question of what happens if the incorrect department code is filled in.

4.1.6 Much of the system is based on the principle of a specified quantity of fuel being dispensed into a vehicle at a specified mileage. However, no account is taken of the fuel in the tank at the beginning or end of the month, since the vehicles are never filled. Consequently at best any consumption figures will only show the relationship between the quantity of fuel dispensed into the vehicle and the distance travelled by that vehicle between the first and last fuel stop in the month. This error could be as much as 25%. See example in Appendix C.

[illegible] Since coupons arrive at the Data Processing Branch in no specific sequence, difficulty is experienced with ensuring that all coupons for all vehicles for a given month are in, especilly in view of the lack of sequence control. It furthermore follows that it is not always possible to be certain that the quantity of fuel in a vehicle at the end of a month is the same as the quantity at the beginning of the following month.

[illegible] The fact that petrol coupons and diesel coupons are housed in different places and controlled by different people very often leads to duplication of work in an attempt to provide complete depot usages to both the depot management and head office. There in fact appears to be no good reason for this other than tradition.

[illegible] Control of fuel taken out in jerrycans for grass cutters and other small machinery is in some instances extremely strict and in others almost non-existant. A small steady loss from this source could be quite substantial.

[illegible] The system of reporting is confused in some areas. Some depots are still submitting forms which were discarded in 1981.

[illegible] A major difficulty with the control of fuel arises because of the continual transferring of vehicles, both inter-depot and interdepartment. A vehicle may for example have been filled in the morning and may then be transferred to another depot or department. The quantity of fuel is nevertheless "debited" to the depot who filled the vehicle.

4.1.12 Loss of fuel in the form of vapour varies but may be as much as 3%. This is very difficult to control since it varies with factors such as weather conditions, tank size, quantity of fuel in tank, etc.

4.1.13 Tank reconcil[illegible] statements are often filled in by keeping a record of the previous days t[illegible] reading and simply writing it in the next da[illegible] The received amount is then added, the issues subtracted and the final dip reading calculated. This, however, shows up as a zero variation on the tank reconciliation and so can be controlled. The point nevertheless remains that subdivision of duties is poor. Furthermore, confusion often arises in other situations. Where a tank has two pumps, readings are sometimes shown against the wrong pump: dip readings are often converted to litres incorrectly or using the table for a different size tank; intertank transfers do not require coupons.

4.2 Data Processing and Reporting

4.2.1 The fact that anyone is entitled to alter the reconciliation sheets without some form of verifying cross check is in itself a loss of control, although it does ensure a stable point from which to compare the data produced.

4.2.2 Differences between the two systems (which occur regularly) are most often caused by incorrect handling of the data rather than inconsistent data. This is inevitable when attempting to manage a situation utilising data from two different systems.

4.2.3 No sequential control is kept from batch to batch, the principle being that if the fuel quantities balance, all the documents must be present.

4.2.4 The parameter cards used in the Fleetmaster system are often out of date and require constant updating to correct information such as department, fleetnumber, etc. This is partially due to the lengthy procedure which must be followed every time a vehicle is transferred, scrapped or purchased and partially to previous punching mistakes made by the data processing entry clerks.

4.2.5 It sometimes occurs that all coupons do not reach the Data Processing Branch before the close-off date. These coupons are punched together with the following months' coupons and will cause a discrepancy between the two systems.

4.2.6 Fleetnumbers are not consistent throughout the department, e.g. fleet type 07 does not necessarily designate the same type of vehicle throughout the department. Furthermore, in some instances, fleet types contain a mixture of vehicle types, making comparisons with fleet averages meaningless.

5. RECOMMENDATIONS

The basic purpose of the entire system is twofold:

To ensure that there are no opportunities for fuel shortages to occur, i.e. to ensure that all fuel purchased by the department is finally utilised in one of the CED vehicles.

To be able to report on various fuel and vehicle statistics in a meaningful manner to allow management to effectively exercise control over fuel usage, vehicle utilisation, cost maintenance, and other aspects of vehicle cost control.

5.1 Fuel Issue and Data Collection

In attempting to prevent fuel shortages the system is to a large extent successful. Nevertheless, it appears that the coupon system needs refinement in some areas.

5.1.1 Issues of coupon books from Mechanical W/S should be entered into a register which should tally with a similar register held in each depot. In order to facilitate control of coupon issues and delegation of responsibility for ensuring that this depot register reconciles with coupons issued, the diesel and petrol coupon books should be kept in the same office. Certainly they should not be kept by the pump attendant. This implies that coupons will be filled in under office supervision rather than at the pump. It also implies that the responsibility and accountability could be delegated to one specific individual. It would furthermore eliminate much of the duplication of work created because of the necessity for the local pump personnel to supply information to depot administrative staff.

5.1.2 Under no circumstances should the person taking the tank readings, also take the pump readings. Furthermore, if two people are taking the readings, they should alternate functions at irregular intervals, although this may be somewhat difficult to achieve in practice. Such malpractices as adding the litres issued from the pump to the morning tank reading to give the closing tank reading, instead of recording the actual closing tank reading, should be dealt with severely.

5.1.3 No coupon or stub should be printed without a preprinted depot and branch code, and the rule of "no coupon, no fuel" should be strictly enforced. This would mean that all fuel and only such fuel indicated on the coupons and stubs belonging to a particular depot would in fact have been dispensed into a vehicle for usage by that depot.

Since the coupons are seldom returned to the transport department of the depot, any summaries required should be made up from the stubs rather than the coupons. This would ensure that only fuel actually dispensed into vehicles in use by the particular depot would be recorded on the summaries.

5.1.4 Jerrycans containing excess fuel from filling grass cutters should remain in the depot since it is impossible to account for this fuel if it is taken out.

5.1.5 An effort should be made to standardise the system of reporting, i.e. ensure that all depots are submitting information which is correct, accurate, and in a simi ar format.

5.1.6 A major area in which control has been overlooked is the fact that the quantity of fuel in the vehicle at critical points is a total unknown.

The philosophy adopted at this time is that any fuel dispensed into a vehicle is considered as being used after being dispensed into the tank.

A much better philosophy is one in which one considers fuel dispensed as taking the place of fuel <u>already</u> used. This means that any fuel dispensed into a vehicle can be specifically allocated since it is now known how that fuel has been used. Some solutions are discussed below:

5.1.6.1 If the vehicle is <u>filled</u> at all critical points, i.e. at the end of the month, before a transfer, on return from a transfer; then the exact quantity of fuel used between that critical point and the last is known.

This is a psychologically important point, since it would avoid depots being billed for fuel not used by them. The practical implementation of this system could be achieved by providing each vehicle with its own designated coupon book which would not have prefixed amounts. The vehicle would always be filled and the quantity of fuel would be inserted in both the coupon and stub. The stub would serve the dual purpose of both stub and logbook in which could be recorded mileage before transfer, and mileage on return from transfer.

Disadvantages of this system are the loss of control over filling in the stub/coupons and also the problem of the coupon books going astray. However, the accuracy with which one could then calculate the fuel consumption of each vehicle would soon uncover irregularities. It would also be a requirement that each logbook be presented to and initialled by the transport superintendant every day, possibly to be held overnight.

5.1.6.2 Each vehicle would only be <u>filled</u> prior to a transfer or month-end. At other times the system would remain as it now exists, and coupons would always be obtained from the depot transport office. The disadvantage with this system is that the stub can only be completed once the vehicle has been filled.

This could lead to irregularities if not controlled properly. However, it is easy to implement and special care could be taken by office staff to ensure correct figures over monthend and prior to transfers.

5.1.6.3 If the quantity of fuel in the vehicle can be accurately read at a given time, the same effect can be achieved, without the necessity of actually filling the vehicle. Most known gauges at this time are not sufficiently accurate for this purpose, although some research is presently being carried out utilising the weight of the fuel as a measure of quantity. This has been found to be sufficiently accurate but many practicalities have yet to be solved.

5.1.6.4 If the quantity of fuel actually passing into the engine can be measured, in terms of time or kilometres travelled, a much more accurate estimate of the fuel consumption can be obtained.

Since the distances are known exactly, the actual fuel used by the vehicle can be computed with a high degree of accuracy.

Unfortunately such electronic systems are expensive and require to be fitted one to each vehicle. They also require a small micro-processor to be able to read the information from the unit situated in the vehicle. The unit in fact plugs into the micro-processor at the end of the day and all the relevant information is passed to it.

It might be pointed out, however, that these systems do provide other much needed information on idling times, excessive speeds and engine revs, brake wear, trip times, etc. One advantage of using such a system is that the existing coupon system could stay largely the same. One example of some of the data available is shown in Appendix D.

5.2 Data Processing and Reporting

The attempt at providing timely, meaningful management information from the system is something of a failure.

Although some possible changes are suggested in section 4.2, the basis of the problem is the nature of the design of the systems being used to process the data. It is evident that the two systems have evolved over a long period of time and have been altered with time to meet the changing needs of the environment. However, with the increasing requirement for "real-time" information, the basic method of functioning of these systems is becoming more and more incompatible with the need for accurate, up to date, readily available and easily understandable management information. Some broad solutions to this dilemma are discussed below:

5.2.1 An entire revision of the two systems at present in use with a view to streamlining their operation and the possibility of combining them into one. A detailed investigation would be necessary to determine how the entire batch loading process could be changed to shorten the time from initial dispensing of fuel into the vehicles to final reports being produced.

5.2.2 The purchase of a more advanced fleetmaster system, which would incorporate all aspects of vehicle cost control, replacement analysis, etc., and which would make more use of online loading of information. This might mean that certain depots would be given terminals where they would enter the information themselves, thus removing the delays of much of the batching process.

Such a system might include such capabilities as sophisticated calculations for loss due to vapourisation if such parameters as mean temperature, tank size, etc. were available. Fuel delivered to a depot could be keyed in at the depot and instantaneously compared with the previous readings and any order information which might be outstanding.

Both of the above solutions are costly, the latter more so than the former, and both would require an in-depth investigation into commercially available systems and their applicability, although it is obvious that this decision cannot be delayed for any length of time. However, such an investigation is beyond the scope of this dissertation. In the short term, however, a simpler solution does exist. One which to a large degree will meet the requirements of readily available, easily understandable and meaningful management information at a small fraction of the cost of the major systems described above. It does have disadvantages, and these will be discussed, but as a short to medium term solution, the system described in the remainder of this paper provides an acceptable solution to the more urgent of the problems described above.

6. THE SYSTEM DESIGN

6.1 Introduction

This stystem has been written on a Hewlett Packard HP 86 microprocessor using interpretive BASIC. A number of reasons account for this choice.

(a) The HP 86 was the only microprocessor readily available, both at the University and at the City Engineers Department.

(b) BASIC is an easily understandable language, is compatible with most microprocessor operating systems and is therefore easily transferable, and is a suitable language for the interactive nature of the system.

The overall objectives of the system are:

(a) simplicity of operation.
(b) Accuracy of data.
(c) Speed of obtaining results.
(d) Meaningfulness and understandability of results.

In designing the system, the following limitations were considered:

(a) Limited storage space (use is made of flexible discs).
(b) Limited system security (interactive BASIC on this machine has inadequate password and file security).
(c) Slow data input rate (no means is available to transfer data from existing magnetic sources onto the micro).
(d) Limited output facilities (output peripherals take the form of a small two pen plotter and a slow printer).

6.2 Use of Quality Control Techniques

One of the simplest, most effective methods of discerning, in a given volume of data, any chance variations which might occur around the mean, from those variations wich have some assignable cause, is the use of well known statistical quality control techniques. These assignable cause variations may then be investigated while others may be ignored, hence saving costly and fruitless investigations. In view of this, and the available output peripherals, it was decided that the system would provide management with a series of accurate, easily understandable statistical graphs which would allow management to assess trends and variances in fuel consumption for a single vehicle type, complete depot or entire branch.

The techniques investigated were:

- Control Charts
- Cumulative Sum Charts
- Moving Averages
- Exponential Smoothing
- Regression Analysis

6.2.1 Control Charts

The Control Chart has a great advantage in that it is visually simple - inspection of the chart can provide a great deal of information. Furthermore, many techniques have been proposed to help interpret any changes which may have taken place. Such techniques as placing limits at 3σ or 2σ away from the mean and determining the probability of a given number of observations falling on the same side of the mean, make interpretation of the chart much easier.

... ...ise ... the sample size does ... remain constant and where the sample size ...roaches 15 - 20. In the former case the ...tution is to change the spread of the ...ontrol limits appropriately while in the ...atter case, the accuracy of the range approximation to the standard deviation is maintained by keeping the sample size low, i.e. keeping the within sample variation small. Note, however, that the distribution of average values from a non-normal universe approaches normality as sample size increases. Since we are unsure of the normalcy of the universe, sample size should not be made too small. Obviously a trade off is necessary. Another advantage of the Control Chart is that large deviations from the mean can be determined far more quickly than with the other methods. Incidentally determination of the actual mean, i.e. not historical mean, is ... ...hat more difficult with this technique.

... ... Charts

By far the greatest advantage of the Cusum Chart is the speed with which a small, sudden and sustained shift in mean can be detected, whereas in the Control Chart, these may be obscured by residual variation. This is because the Cusum utilises a combination of successive results whereas the Control Chart measures each observation independently.

...tection of exactly at what point a change has occurred is visually easy with a Cusum Chart, although these changes are most accurately determined when they occur around the mean (horizontal) since changes in slope are more easily measured in slopes of less than, say, 60°. Furthermore, Cusums can, if not scaled correctly, be visually difficult to interpret due to their tendency to "run" off the page. Experience has shown that a realistic scale is one unit on the horizontal scale representing 2σ units on the vertical scale, although in this case, physical considerations have determined the scales used. One other advantage is that the Cusum Chart may be used to indicate trends in data, although other methods such as Moving Average Charts show these as well, and with greater flexibility.

6.2.3 Moving Average Charts

Very often the most important requirement in measuring past data is to determine, with reasonable accuracy, any trends in this data. This is best achieved by plotting a Moving Average graph of the data.

The intention in plotting these trends may be to establish whether some goal has been met in the past, to forecast from the present trend what future data may look like, or simply to determine whether any trend exists at all. In any event, the period over which the moving average is calculated will determine to what extent unwanted fluctuations are eliminated - the larger the period the smoother the curve. The disadvantage of spreading the period too wide is the increasing lack of accuracy with increasing spread. Once again the situation calls for a trade off, the usual spread adopted being 12 months.

6.2.4 Other Techniques

Techniques using a combination of moving average graphs and control charts, as well as minor variations of each of the above techniques have been used in certain areas of quality control. The disadvantage of these methods is the added complexity they introduce, and they have not been considered when choosing the most suitable techniques for this system.

Other techniques such as exponential smoothing and regression analysis have been found extremely useful for forecasting future trends from existing and past data. However, since this is not the primary purpose of the system, these techniques have not been incorporated in the system design.

Furthermore, in evaluating the advantages of the above three methods, it was decided that in view of the widely varying circumstances, none would provide sufficient information on their own. Consequently the system was designed utilising all three techniques.

Introduction of the system would probably best be achieved by initially directing the attention of operations personnel to the more easily understandable control chart and that of management to the trend-like moving averages. Once these have been accepted, the cusum charts could be incorporated allowing more accurate interpretation of the available techniques.

6.3 Indexed File Simulation

It is common on most microprocessors that only sequential or relative file access is supported. In order to be able to retain the existing vehicle coding system but to allow direct access to data for any vehicle type of any depot, two data files have been created for each depot, one being a tag file permitting relative access according to vehicle type, and containing as data, the relative address of the main data for that vehicle type. This in fact simulates the operation of an indexed file organisation. An example is shown in Appendix E.1.

6.4 File Size Consideration

Due to the nature of file organisation on a flexible disc, it was calculated that far more space would be required if a single file per branch was created than for a file per depot. Furthermore files can be split onto more than one diskette more easily using the latter technique. The disadvantage of this technique is the comparative difficulty of controlling and securing numerous files as opposed to one. An example of file space calculation is shown in Appendix E.2.

6.5 Menu

The system is initialised simply by inserting the master diskette into the diskette drive prior to power up. On power up the master diskette is searched for the start routine which then allows access to the main menu.

It was felt that a menu system is easier to understand, is more user friendly and is faster than a system which requires the user to load each job as required.

6.6 System Breakdown

A brief discussion detailing the system breakdown follows. A more detailed analysis of the system features together with the relevant flow charts can be found in Appendix H.

6.6.1 Input

All data is entered via the keyboard, although in the longer term the possibility exists of copying the information from the existing fleetmaster system, onto magnetic media and transferring it to this system. In the short term, however, the more laborious manual method is unavoidable.

The following functions are used to input data to the system:

6.6.1.1 Data Load

Monthly data for each vehicle is loaded and combined into vehicle types per branch-depot combination. New vehicle types, depots and branches may be loaded, and previous data viewed.

6.6.1.2 Control Parameter Input

System Control Date, Moving Average Spread and Control Chart Limit Factor may be entered or modified using this program.

6.6.1.3 Default Parameter Input

Predetermined defaults for including or excluding vehicle types or depots in a consolidation of data as well as to identify types as petrol/diesel are loaded with this program.

6.6.1.4 Branch/Depot Load

This program is used to enter or modify the description of a branch and/or depot. Any alphanumeric names may be used.

6.6.1.5 Coefficient of Range

The statistical table containing Hartleys coefficients with which to determine the physical spread of the statistical control chart limits utilising the range, is loaded with this program. Once loaded, these should not require to be changed.

6.6.1.6 Load Mean/Range

This program is used to load the historical mean and range for a vehicle type, depot or branch. Note that the range for depots and branches are not summations of individual types due to the fact that the vehicle types included are not uniform.

6.6.2 Processing

A number of different techniques are used to provide useful information about the data loaded. They are complimentary to one another and should be used together if any great degree of certainty is to be achieved in making decisions based on this information.

6.6.2.1 Simple Data Plots

This program plots the data points for the chosen combinations. Actual fuel used can be determined from this graph.

6.6.2.2 Moving Averages

Data for the given combinations for the past 3 years are accumulated and averaged over the moving average period, n, which may vary between 2 and 24 months (see H.1.2.2). Obviously the first n months are not printed since no average is available for them. Trends in data are easily recognised from these graphs.

6.6.2.3 Control Charts

The program calculates both a mean and a range chart for the chosen branch-depot combination. Historical mean and range are taken from the history file and control limits are calculated utilising these mean / range figures and the coefficients loaded in the statistical table.

6.6.2 Processing

A number of different techniques are used to provide useful information about the data loaded. They are complimentary to one another and should be used together if any great degree of certainty is to be achieved in making decisions based on this information.

6.6.2.1 Simple Data Plots

This program plots the data points for the chosen combinations. Actual fuel used can be determined from this graph.

6.6.2.2 Moving Averages

Data for the given combinations for the past 3 years are accumulated and averaged over the moving average period, n, which may vary between 2 and 24 months (see H.1.2.2). Obviously the first n months are not printed since no average is available for them. Trends in data are easily recognised from these graphs.

6.6.2.3 Control Charts

The program calculates both a mean and a range chart for the chosen branch-depot combination. Historical mean and range are taken from the history file and control limits are calculated utilising these mean / range figures and the coefficients loaded in the statistical table.

Due to space considerations, it is not possible to hold actual data for each vehicle or mean/range data for every combination of vehicle types. Consequently in constructing range charts for depot totals and branch totals the range has been reconstructed using an approximation (See H.1.2.3). Extreme care should therefore be taken when making decisions based exclusively on these range charts. Furthermore, due to the second constraint, only certain combinations of totals may have control charts, i.e. per vehicle type; a series of vehicles; total of all default types within a depot; total of all default types within a branch. Ad hoc combinations are not allowed.

This technique should be used when investigating specific data points with respect to the mean and range. It is more complex than the previous technique and should be introduced with care.

6.5.2.4 <u>Cusum Charts</u>

The cumulative sum is calculated for the chosen combination utilising the actual data and the stored historical mean. An example showing how the change in fuel usage is represented by the slope of the curve is given in Appendix G.

Existing statistical tests for this technique are somewhat complex, and even the approximation described in H.1.2.4 is difficult to implement on a day to day basis. Consequently the practice has been adopted of simply evaluating the graph critically to determine whether any change is sufficient to warrant investigation.

As in the control chart, cusum charts may only be printed for certain combinations due to the same constraint on space. This technique concentrates on the relative change in usage over consecutive months and together with the control chart, provides an excellent measure of statistical control.

6.6.2.5 File Roll

Since space provides for only 3 year's data to be retained, a roll facility exists to drop off the first month and roll each month back by one. This function must be performed each month prior to loading new months data. The roll may be performed on all depot-branch combinations together or individually as preferred by the operator.

6.6.3 Output

Output from the system consists mainly of graphical representation of chosen data.

6.6.3.1 Simple Data Graphs

These graphs are produced as determined by the request in 6.6.2.1. Limits and scales are determined from the data itself thus obviating the need for operator intervention.

6.6.3.2 Moving Average Charts

Plotted as requested in 6.6.2.2, these graphs exclude the initial n months of data, since this period has no moving average associated with it.

6.6.3.3 Control Chart Plots

These plots display only two years historical data, since it is necessary to plot the mean and range chart one below the other. This is considered sufficient data to give an accurate picture of the current situation.

6.6.3.4 Cusum Plots

Although suggestions are made in the literature (see H.1.3.4) regarding axis scales, the dimensions of the plotter required that the scale be allowed to vary with the nature of the data. A key is provided showing registration of major angles on the chart.

6.6.3.5 Print of Data

This produces a listing detailing, for the chosen combination, consumption figures per month as well as year-to-date. Petrol and Diesel are listed separately.

6.6.3.6 Print of Depots/Branches

A listing of all existing depots and branches with their descriptions may be printed on request.

SYSTEM OPERATION

A worked example of each of the programs in the system together with applicable remarks is given. Where an option is given, only the most commonly used option is shown, (in parentheses), although some reference will be made to the other choices in the remarks.

The system is menu driven and all programs link back to the main menu on completion. The main menu is shown in fig. 7.0

[illegible]

FIG 7.0

[illegible] options will now be discussed.

Load Data

This program [illegible] [illegible]iated [illegible] [illegible] [illegible] [illegible] function
[illegible]).

[illegible]

[illegible]

(Ø1)

[illegible]

(Ø1)

[illegible]
[illegible]

(Ø1)

[illegible]
[illegible]
[illegible]

(Y)

FIG 7.1.1

In fig. 7.1.1, branch, depot and type must be numeric
The answer No (N) returns the program [illegible] [illegible]
beginning.

[illegible] (N)

FIG 7.1.2

Here the program validates the file status. If the answer does not agree with the scan of the disc directory, an error message is printed and the program returns to the beginning.

[illegible] (Y)

[illegible]
[illegible] (1)

[illegible]
[illegible]
[illegible]

[illegible] (N)

FIG 7.1.3

In fig 7.1.3, the option to display all data will display the last 36 months of data for this vehicle type in this depot-branch combination. The option not to display any data advances the program to fig 7.1.4.

[illegible] (Y)

FIG 7.1.4

The option (N) in fig 7.1.4 would result in a warning message informing the user that unless the file is rolled first, information will be overwritten. This warning may be overridden if the user so desires.

[illegible]	(Y)
[illegible]	

FIG 7.1.5

Only numerics from 1 to 36 will be accepted. As in the case of the display, the oldest month is designated 36, the youngest is 1.

ENTER VEHICLE FLEETNO	(Ø1)
ENTER FUEL USED	(5000)
ARE THERE MORE VEHICLES [illegible]	(Y)

FIG 7.1.6

In fig 7.1.6 the series of 3 questions is repeated until there are no more vehicles in that type. The fleetnumber must be a four digit numeric and fuel used may not be negative.

[illegible]	(N)
[illegible]	

FIG 7.1.7

Replying (Y) to the question in fig 7.1.7 returns the program to fig 7.1.1.

After storing of all data, the program returns to the main menu.

7.2 Control Parameter Maintenance

This program is initiated from the main menu (function 9).

```
                    [illegible]

                    1. [illegible]
                    2. [illegible]
                    3. [illegible]

[illegible]
                                        (3)
```

FIG 7.2.1

Selection of (1) and (2) would result in an echo of the control limit or the control date.

```
[illegible]
                    [illegible]
[illegible]
                                        (12)

[illegible]
                    [illegible]
[illegible]
                                        (Y)

                    [illegible]
```

FIG 7.2.2

Any period within the bounds of the discussion in Appendix H.1.1.2 is acceptable.
The answer (N) in fig. 7.2.2 returns the program to the beginning of fig. 7.2.2
After storing of the data, the program returns to the main menu.

7.3 Default Maintenance

This program is initiated from the main menu (function 12).

```
(Ø1)

(Ø1)

(Ø5)

(Y)
```

FIG 7.3.1

The answer (N) returns the program to the beginning.

```
(Y)
```

FIG 7.3.2

The answer (N) in fig. 7.3.2 terminates the program which then returns to the main menu.

[illegible]
(N)

[illegible]
(N)

[illegible]
(N)

[illegible]
(D)

[illegible]

[illegible]

[illegible]
(Y)

FIG 7.3.3

The answer (N) returns the program to the first question in fig. 7.3.2

[illegible] (N)

[illegible]

FIG 7.3.4

The answer (Y) returns the program to the beginning of section 7.3 after storing the change already affected. The answer (N) stores the data and returns to the main menu.

7.4 Branch/Depot Load

This program is initiated from the main menu (function 7/8). Since these two functions are identical, only the branch load is described below.

```
[illegible]

[illegible]
                                                       (Ø1)

[illegible]
1

[illegible]
                                                       (CLEANSING)

[illegible]
[illegible]
[illegible]
                                                       (Y)

[illegible]
```

FIG 7.4.1

Numeric validation is performed on the code of the branch which must be between 1 and 50. No validation is performed on the description loaded. The answer (N) to the echo causes the program to return to the request for the new description. The answer (Y) results in storing of the new description on the branch file and the program returns to the main menu.

Note that no descriptions may be loaded for a vehicle type. This is due to the fact that these types are not standard and are simply referred to numerically.

7.5 Coefficient of Range Load

This program is initiated from the main menu (function 10).

```
[illegible]

[illegible]
                                                    (2)
[illegible]

[illegible]
[illegible]
[illegible]

[illegible]
                                                    (Y)
```

FIG 7.5.1

The groupsize entered must be numeric. The response (N) returns the program to the first request in fig. 7.5.1

```
[illegible]
                                                    (1.88)
[illegible]
                                                    (Ø)
[illegible]
                                                    (3.27)
[illegible]

[illegible]
[illegible]
[illegible]

[illegible]
                                                    (Y)
[illegible]
                                                    (N)
[illegible]
```

FIG 7.5.2

All factors must be numeric. The response (N) to the echo returns the program to the first request in fig. 7.5.2. The request (Y) to change other factors returns the program to the first request in fig. 7.5.1. The response (N) causes the data to be stored in the statistics table and the program returns to the main menu.

7.6 Mean/Range Maintenance

This program is initiated from the main menu (function 6).

```
                    *** MEAN/RANGE MAINTENANCE ***
                    YOU CAN MAKE THE FOLLOWING CHANGE
                    1.CHANGE MEAN RANGE OF A FILE
                    2.CHANGE MEAN RANGE OF A DEPOT
                    3.CHANGE MEAN RANGE OF A BRANCH
                    4.END
MAKE YOUR SELECTION
                                                (1)
WHICH BRANCH
                                                (Ø1)
WHICH DEPOT
                                                (Ø1)
IS THIS A NEW FILE Y/N
                                                (N)
```

FIG 7.6.1

In fig. 7.6.1 the program validates the branch-depot combination and returns an error message if the answer is not in agreement with the scan of the disc directory, e.g. if such a branch-depot combination does not exist.

```
[illegible]
                                        (Ø1)
[illegible]
              [illegible]
              [illegible]
[illegible]
                                        (1100,68)
[illegible]

              [illegible]
              [illegible]
[illegible]
                                        (Y)
```

FIG 7.6.2

The answer (N) in fig 7.6.2 returns the program to the beginning of fig. 7.6.2 and the type must be re-entered.

The options (2) and (3) in fig. 7.6.1 result in a request for the branch and depot in the former case, and for the branch only in the latter case.

After storing the data the program returns to the first question in 7.6.1. The option (4) will then return the program to the main menu.

Simple Data Plots

This program is initiated from the main menu (function 2)

```
[illegible]
```

FIG 7.7.1

All variable initialization is performed in this step. No operator input is required and this step takes approximately 15 seconds.

```
[illegible]
                                              (1)
```

FIG 7.7.2

Only the numeric responses 1, 2 or 3 are acceptable.

```
[illegible]
                                              (01)
[illegible]
                                              (0)
```

FIG 7.7.3

Each depot is stored in a table and on completion of input a validation test is run to ensure that all branch-depot combinations are valid. Any invalid combination causes a message to be displayed on the screen. Multiple branches may not be loaded, and multiple depots may only be loaded for choice 3.

```
TYPES TO INCLUDE
ENTER CODE OR 0 TO END [illegible]                       (01)

MORE TYPES
ENTER CODE OR 0 TO [illegible]                           (0)
```

FIG 7.7.4

Any number of types up to 10 may be included. When attempting to enter the 11th, the system requests the user to print the first batch of 10 graphs and then re-input from the 11th onwards. If the default is requested (100), it must be entered before any other individual type, failing which the system issues a message to the effect that default and specific requests may not be mixed. No restriction exists on combinations of branch-depot-type requests.

```
[illegible]                [illegible]

                           [illegible]
                           [illegible]
                           [illegible]
```

FIG 7.7.5

The system indicates the file name and the disc on which it is located. As each type is processed, this fact is indicated by the system.

The program now hands control to the plotting program:

```
[illegible]
PLEASE SET UP [illegible] (C)
[illegible]
```

FIG 7.7.6

Entering E here returns the program to the main menu without the plotting function being performed. If the plotter is not switched on, the system will request the operator to switch it on before proceeding.

After plotting the graph, the program returns to the main menu. Examples of the various graphs are shown in the section "System Output" under 7.15.1.

7.8 Moving Average Graphs

This program is initiated from the main menu (function 3).

(1)

(Ø1)

(Ø1)

(Ø1)

(Ø)

(C)

FIG 7.8.1

This format is similar to that described in the previous section. All validations and options are identical. One difference lies in the fact that only 24 months data are printed on these graphs and not 36. This is due to the fact that within the first 12 months insufficient data exists to produce a 12 month moving average. After plotting the graph the program returns to the main menu. Graphs showing examples of the various options may be found under "System Output" in section 7.15.2.

7.9 Control Charts

This program is called from the main menu (function 4).

```
[illegible]

[illegible]
[illegible]

[illegible]

[illegible]

[illegible]

[illegible]
[illegible]

[illegible]                                   (1)
                                              (Ø1)
[illegible]
[illegible]
                                              (Ø1)
[illegible]
[illegible]
                                              (Ø1)
[illegible]
[illegible]
                                              (Ø)
                      [illegible]
[illegible]
                      [illegible]
                      [illegible]
                      [illegible]

                      [illegible]

[illegible]
                                              (C)
                      [illegible]
```

FIG 7.9.1

Once again this layout is similar to that of section 7.7 and all validations in this program are identical to that of 7.7. The major differences in this program are:

- when requesting (2) or (3) in answer to the first question in 7.9.1, no option is given with regard to which types are entered. The program assumes all default types for the requested branch-depot combination.
- to facilitate better comparison, these charts are plotted across the page and not down the page. Consequently only 24 months data is given and not 36. After plotting the charts the program returns to the main menu. Examples of the various charts are shown in the section "System Output" under section 7.15.3.

7.10 Cumulative Sum Charts

This program is called from the main menu (function 14).

```
*** CUMULATIVE SUM CHARTS ***

    PROGRAM BEGINS
INITIALISATION [illegible]

 [illegible] THE FOLLOWING:

1.PRINT THE CHARTS FOR A [illegible]

 .PRINT THE TOTAL [illegible]

 .PRINT THE TOTAL [illegible] THE BRANCH
MAKE YOUR SELECTION
                                                  (1)
WHICH BRANCH
                                                  (01)
WHICH DEPOT TO INCLUDE
(TYPE CODE OR 0 TO END)
                                                  (01)
TYPES TO INCLUDE
[illegible]
                                                  (01)
MORE TYPES
[illegible]
                                                  (0)
                    [illegible]
[illegible]
                    [illegible]
                    [illegible]
                         [illegible]

                    [illegible]
[illegible]
                                                  (C)
                    [illegible]
```

FIG 7.10.1

In Fig 7. 0.1 it can be seen that the standard format has once again been used. As in the case of section 7.9, selection of (2) or (3) in answer to the first question above, results in all default types being accumulated for the requested branch-depot combination.

After plotting the charts, the program returns to the main menu. Examples of some charts are shown in the section "System Output" in 7.15.4

7.11 File Roll

This function is initiated from the main menu (function 11).

```
                    *** DEPOT ROLL FUNCTION ***
              DO YOU WISH TO DO ONE OF THE FOLLOWING:
              1. ROLL ONE DEPOT [illegible]
              2. ROLL ALL DEPOTS FOR A GIVEN BRANCH
              3. END
PLEASE ENTER YOUR SELECTION
                                         (1)
```

FIG 7.11.1

Selecting option (2) in fig. 7.11.1 results in all depots of the branch requested in fig. 7.11.2 being rolled.

```
WHICH BRANCH
                                         (Ø1)
WHICH DEPOT
                                         (Ø1)
```

FIG 7.11.2

If option (2) in fig. 7.11.1 had been selected, the second question in fig. 7.11.2 would not have been asked.

```
PROCESSING TYPE 1
COMPLETED TYPE 1
TYPE 2 DOES NOT EXIST
TYPE 3 DOES NOT EXIST
TYPE 4 DOES NOT EXIST
PROCESSING TYPE 5
COMPLETED TYPE 5
TYPE 6 DOES NOT EXIST
TYPE 7 DOES NOT EXIST
TYPE 8 DOES NOT EXIST
TYPE 9 DOES NOT EXIST
PROCESSING TYPE 10
COMPLETED TYPE 10
TYPE 11 DOES NOT EXIST
TYPE 12 DOES NOT EXIST
TYPE 13 DOES NOT EXIST
TYPE 14 DOES NOT EXIST
TYPE 15 DOES NOT EXIST
  |        |      |     |
  |        |      |     |
  |        |      |     |
TYPE 96 DOES NOT EXIST
TYPE 97 DOES NOT EXIST
TYPE 98 DOES NOT EXIST
TYPE 99 DOES NOT EXIST
```

FIG 7.11.3

Each type is rolled individually, and this fact is indicated on the screen. When type 99 has been rolled, the program returns to fig. 7.11.1. Selection of (3) now returns the program to the main menu. Selection of (1) and (2) will advance the program once more to fig. 7.11.2.

7.12 Consumption Data Print

This program is called directly from the main menu (function 5).

```
[illegible]

[illegible]
[illegible]

[illegible]

[illegible]

[illegible]
                                   (1)
```

FIG 7.12.1

Selection of (2) or (3) in fig. 7.12.1 will result in only the first question of fig. 7.12.2 in the latter case and in the first two questions in the former case, i.e. in both these cases, the program assumes the default types. Selection of (4) returns the program to the main menu. No other input will be accepted.

```
[illegible]
                                   (01)
[illegible]
                                   (01)
[illegible]
                                   (01)
[illegible]
[illegible]
                                   (0)
[illegible]
```

FIG 7.12.2

[illegible]. Depot and type must all be numeric. Up to 10 types may be requested if the selection in fig. 7.12.1 was [illegible]. The program then returns a message to inform the operator that any further requests must be entered after the printing of the current series. The program then prints data for each type, each on a new page. Page numbers are given to prevent the accidental loss of a page.

When printing totals of a depot or of a branch, the system shows a "this month" total and a "year to date" total for petrol types and diesel types separately, [illegible] colums of data. On completion of the print, the program returns to fig. 7.12.1. An example of the data printed is shown in "System Output" in section [illegible].

7.13 Print Branch/Depot Code List

This program is called from the main menu (function 13).

```
[illegible]

YOU [illegible] DO ONE OF THE [illegible]

1. LIST THE [illegible] OF [illegible]

2. [illegible]
MAKE YOUR SELECTION
                                        [illegible]
```

FIG 7.13.1

All branches and depots have been initialised with description "NONE". Thus any depots/branches whose descriptions have not been subsequently loaded will be designated by the above description.

The program now returns to the main menu.

An example of a table of branch codes is shown in section 7.15.6 of "System Output"

7.14 End

This function (function 15) exits from the main menu and signs off the system, giving the date and time signed off, for accounting purposes. Once this function has been keyed, the machine must either be switched off and then back on, or the program Autost must be loaded, before the main menu will once more appear on the screen.

7.15 System Output

These graphs have been grouped together in this section to facilitate comparison between the various types. They are based on a fairly representative sample of test data and should not be misconstrued as actually applying to the Branch-Depot combination designated.

The following table shows a summary of the examples included in the section indicating the figure number and the content of each.

TABLE 7.15.1

Fig	Content				
7.15.1.1	Simple Data Plot	Type 1	Depot Selby	Branch Cleansing
7.15.1.2	Simple Data Plot	Type 1,5,10	Depot Selby	Branch Cleansing
7.15.1.3	Simple Data Plot	Type All	Depot All	Branch Cleansing
7.15.2.1	Moving Average Plot	Type 1	Depot Selby	Branch Cleansing
7.15.2.2	Moving Average Plot	Type 1,5,10	Depot Selby	Branch Cleansing
7.15.2.3	Moving Average Plot	Type All	Depot All	Branch Cleansing
7.15.3.1	Control Charts	Type 1	Depot Selby	Branch Cleansing
7.15.3.2	Control Charts	Type All	Depot Selby	Branch Cleansing
7.15.3.3	Control Charts	Type All	Depot All	Branch Cleansing
7.15.4.1	Cusum Charts	Type 1	Depot Selby	Branch Cleansing
7.15.4.2	Cusum Charts	Type All	Depot Selby	Branch Cleansing
7.15.4.3	Cusum Charts	Type All	Depot All	Branch Cleansing
7.15.5	Consumption Data Print	Type All	Depot Selby	Branch Cleansing
7.15.6.1	Branch Description Table			
7.15.6.2	Depot Description Table			

TOTAL FUEL CONSUMPTION

BRANCH: CLEANSING

DEPOT : SELBY (PETROL)

DEC 1983

VEHICLE
TYPE
1

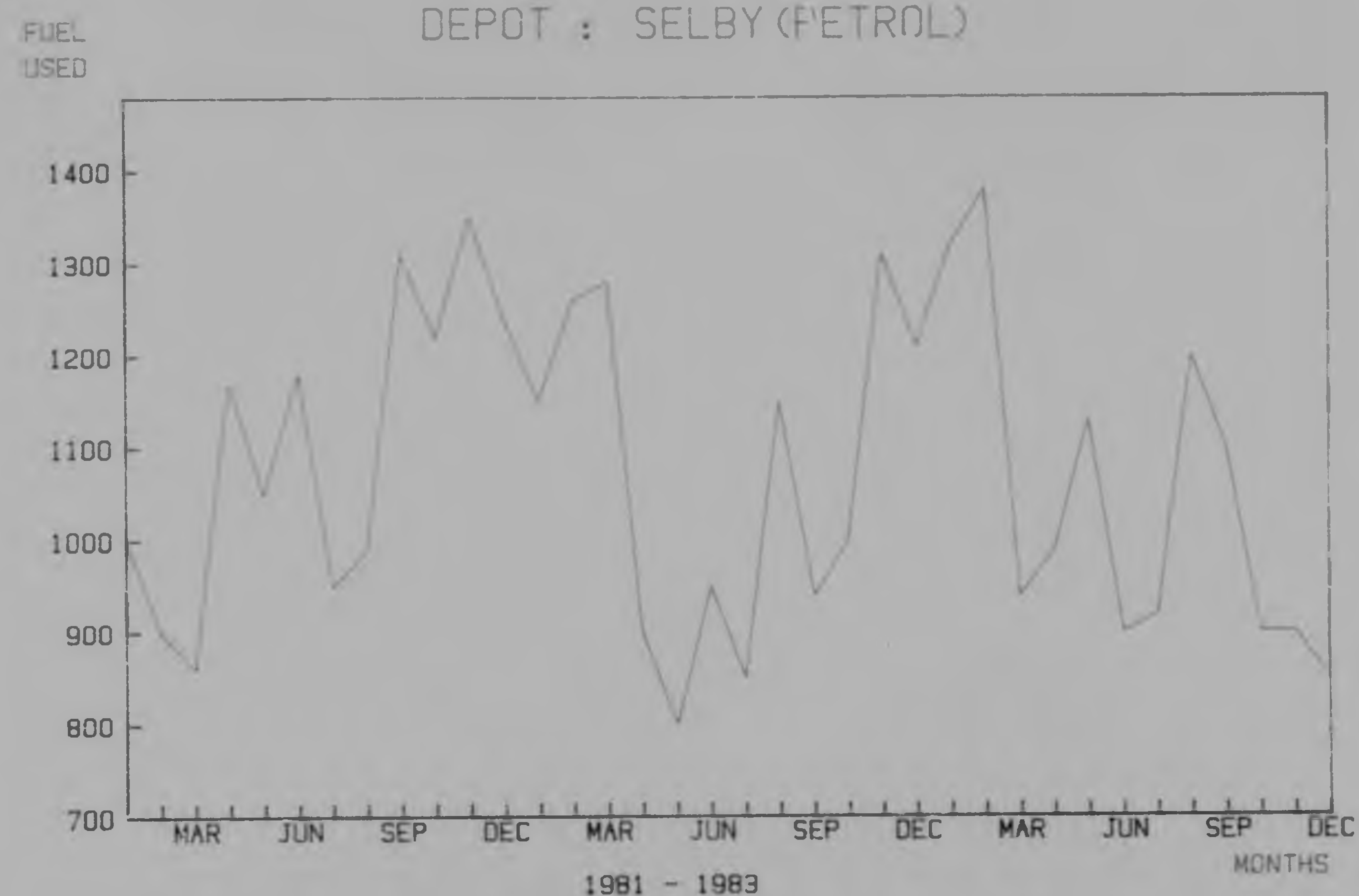


FIG 7.15.1.1

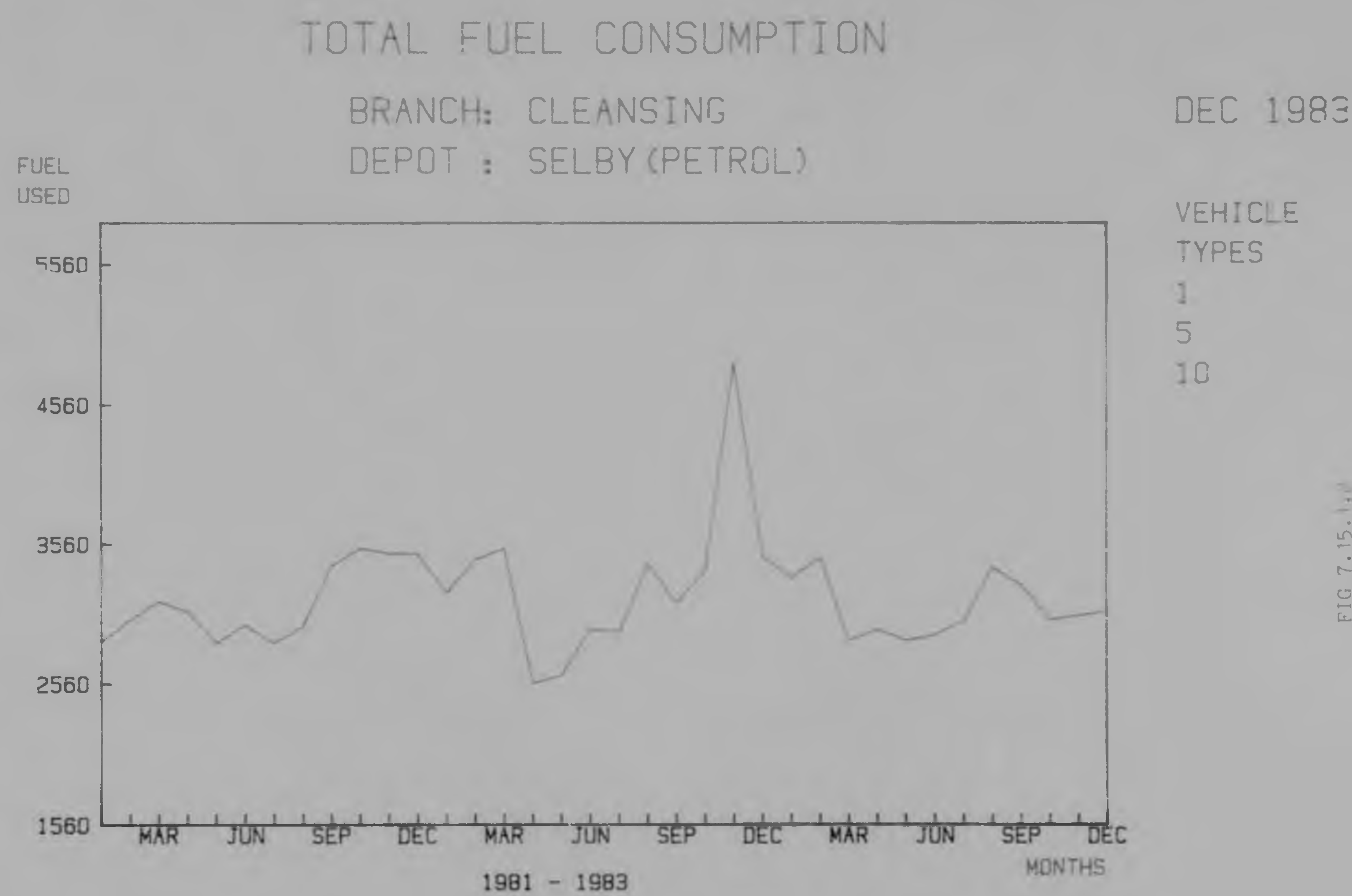

TOTAL FUEL CONSUMPTION
BRANCH: CLEANSING
DEPOT : SELBY (PETROL)
DEC 1983
FUEL USED
5560
4560
3560
2560
1560
MAR
JUN
SEP
DEC
MAR
JUN
SEP
DEC
MAR
JUN
SEP
DEC
MONTHS
1981 - 1983
VEHICLE TYPES
1
5
10

FIG 7.15.

TOTAL FUEL CONSUMPTION

BRANCH: CLEANSING

DEPOT : ALL (PETROL)

DEC 1983

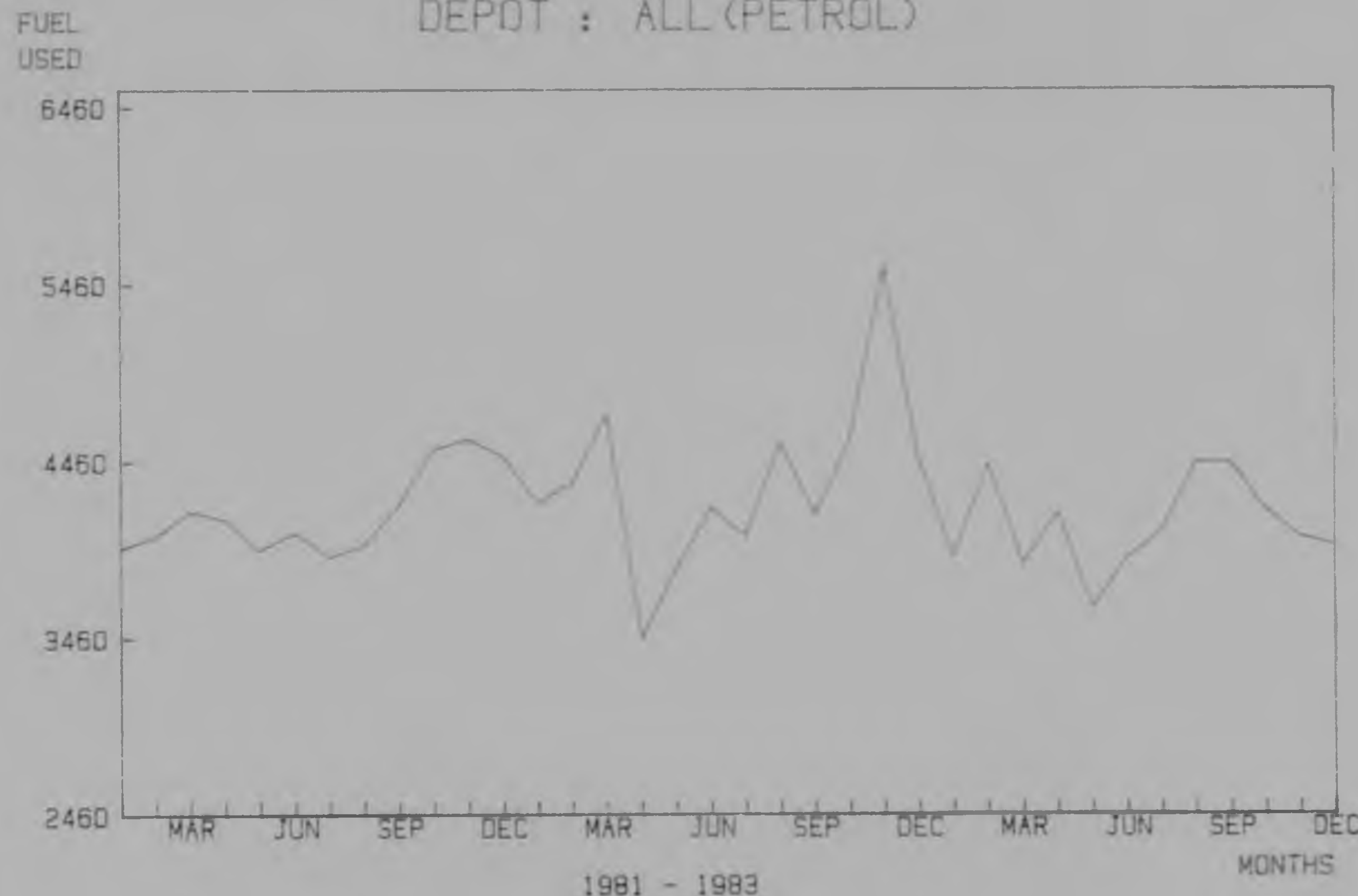


FIG 7.15.1.1

MOVING AVERAGE (12 MONTHS)

BRANCH: CLEANSING
DEPOT : SELBY (PETROL)

DEC 1983

VEHICLE
TYPE
1

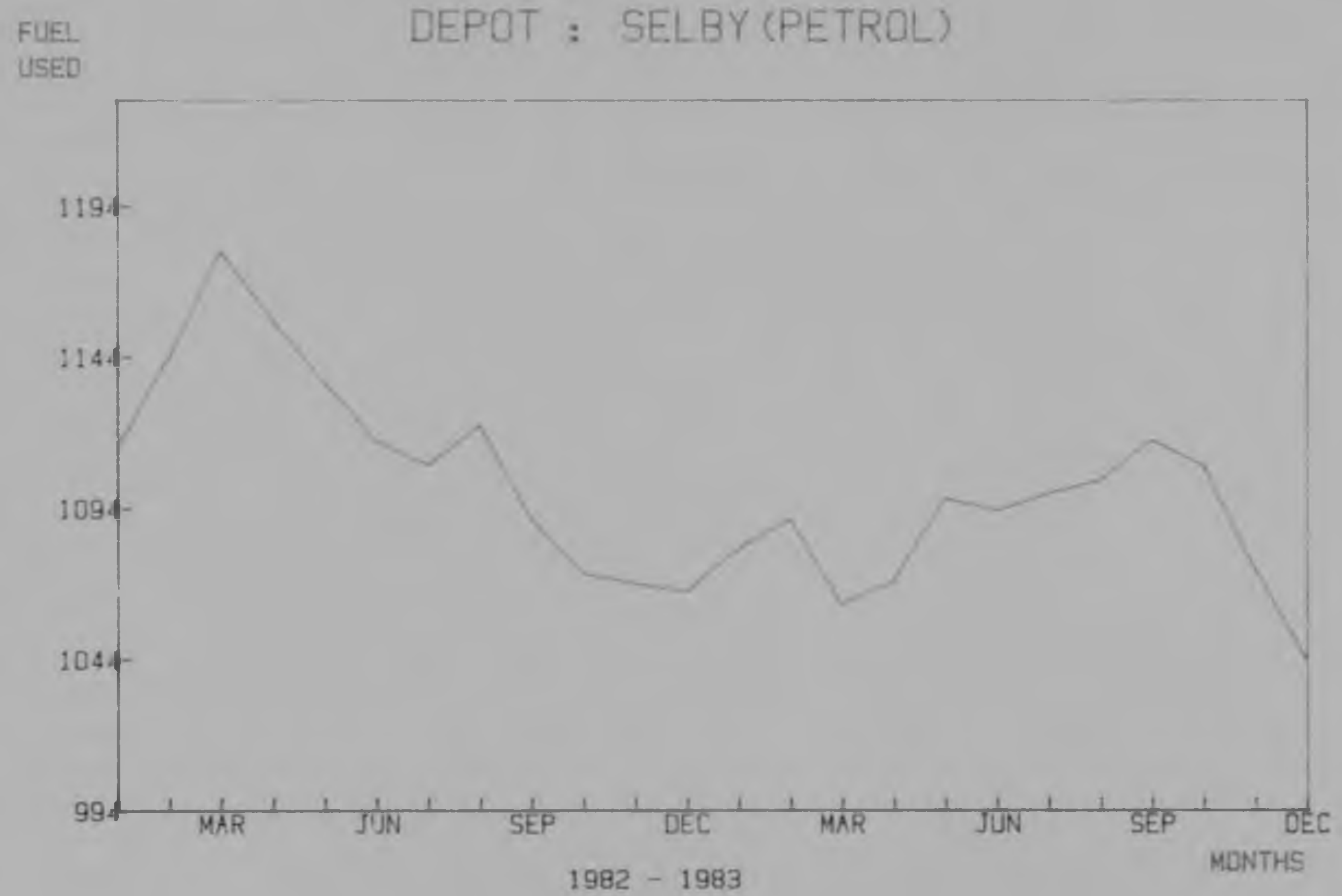


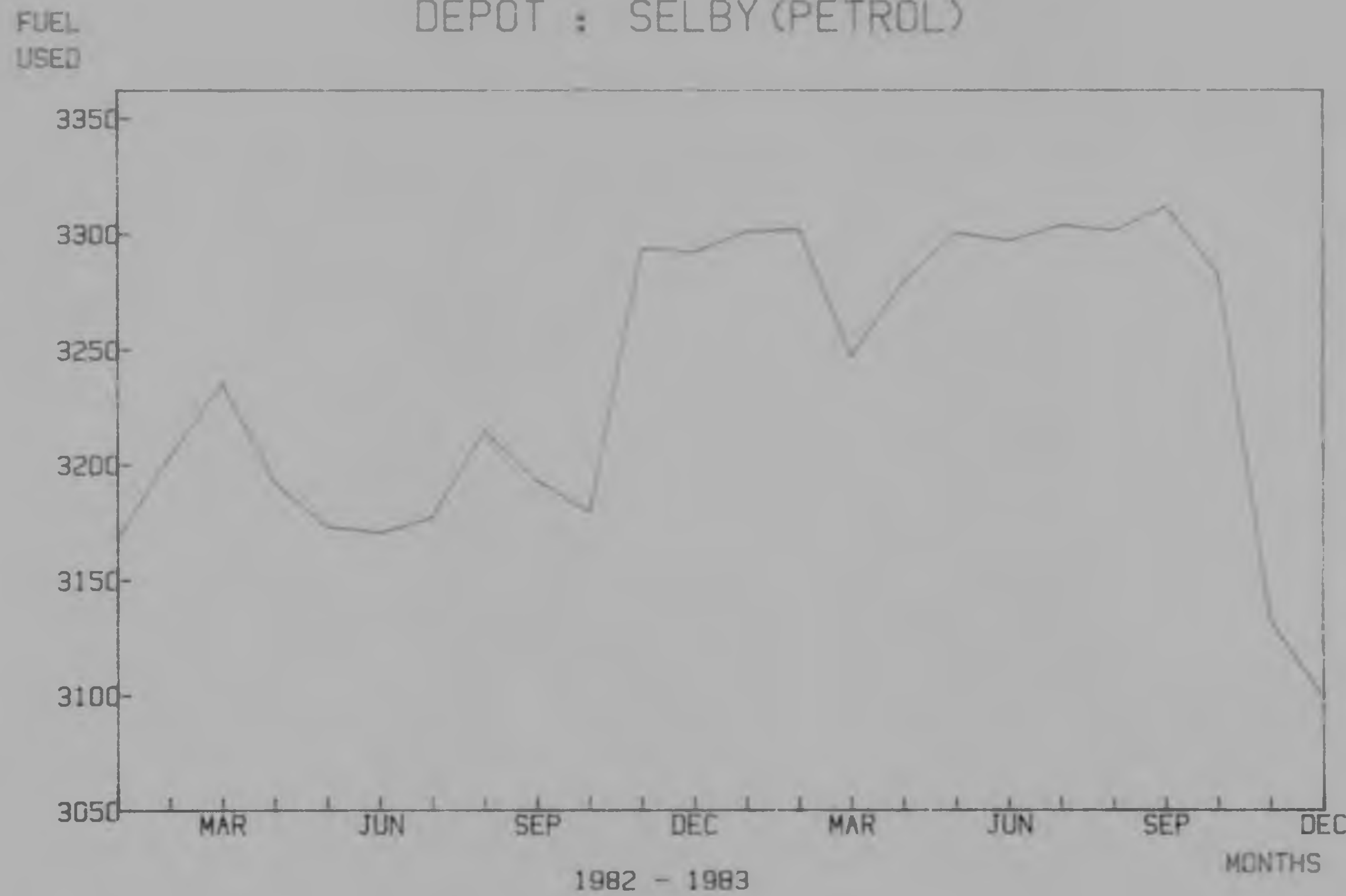
FIG 7.15.2.1

MOVING AVERAGE (12 MONTHS)

BRANCH: CLEANSING

DEPOT : SELBY(PETROL)

DEC 1983



VEHICLE
TYPES
1
5
10

FIG 7.15.2.2

MOVING AVERAGE (12 MONTHS)

BRANCH: CLEANSING

DEPOT : ALL (PETROL)

DEC 1983

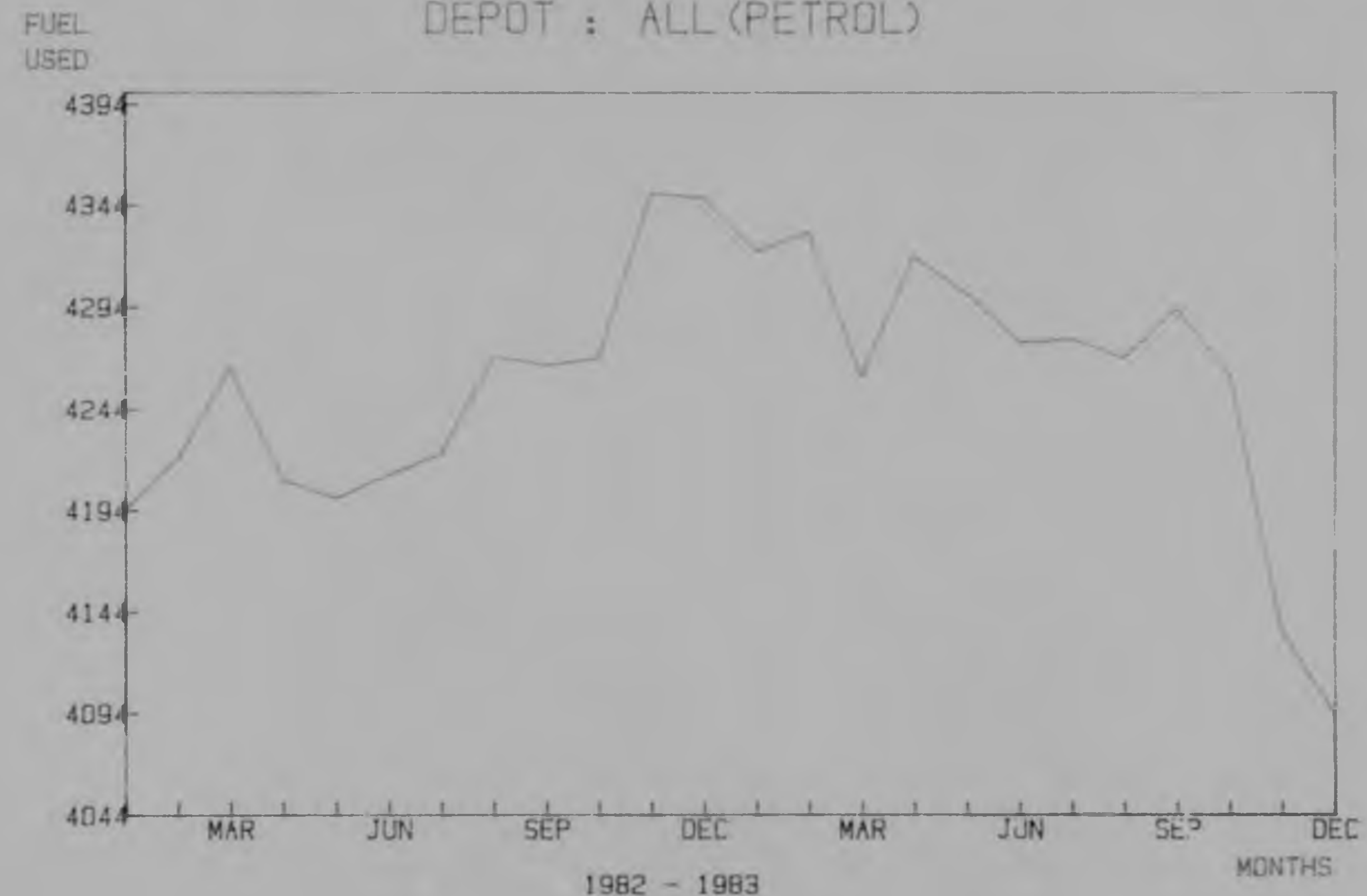


FIG 7.15.2.1

CONTROL CHART (3 SIGMA)

BRANCH: CLEANSING DEC 1983

DEPOT : SELBY (PETROL)

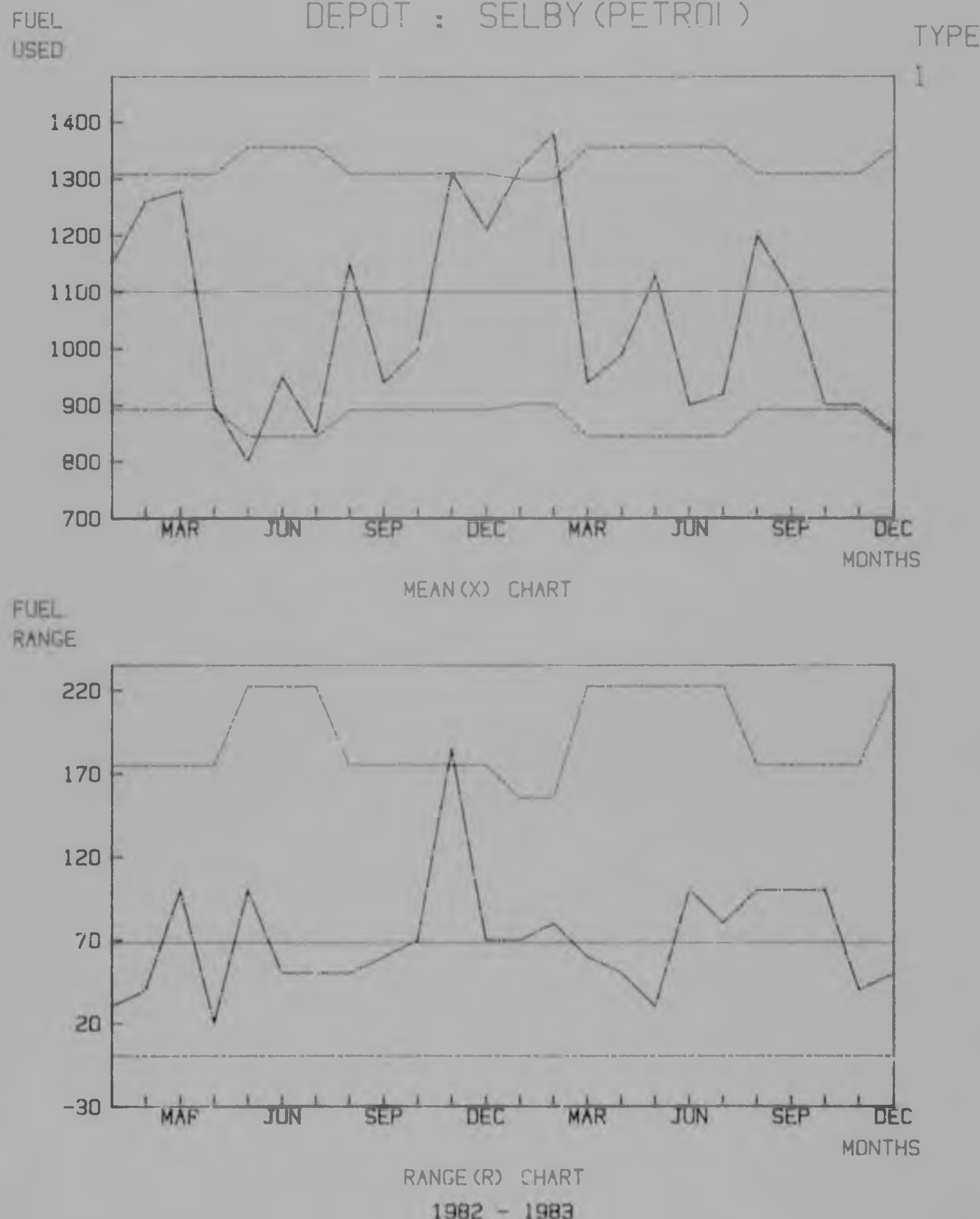


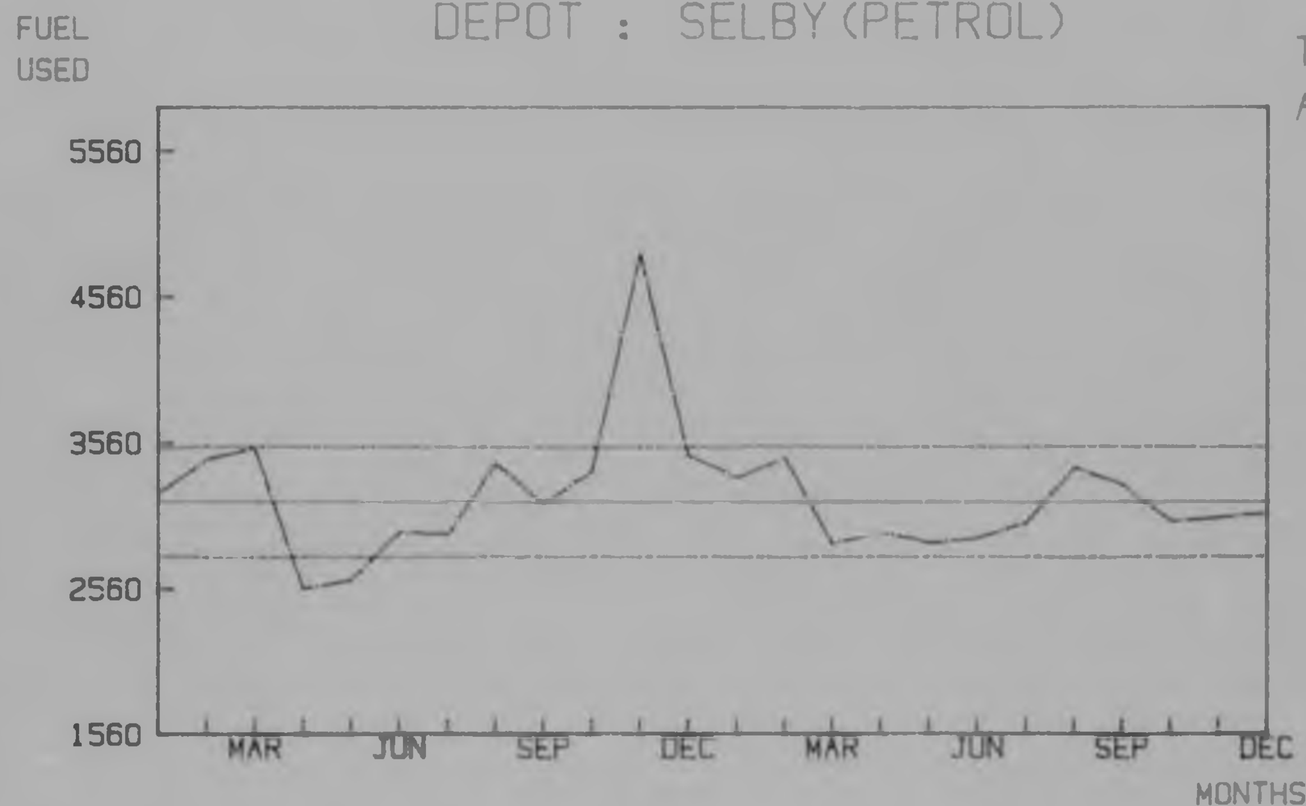
FIG 7.15.3.1

CONTROL CHART (3 SIGMA)

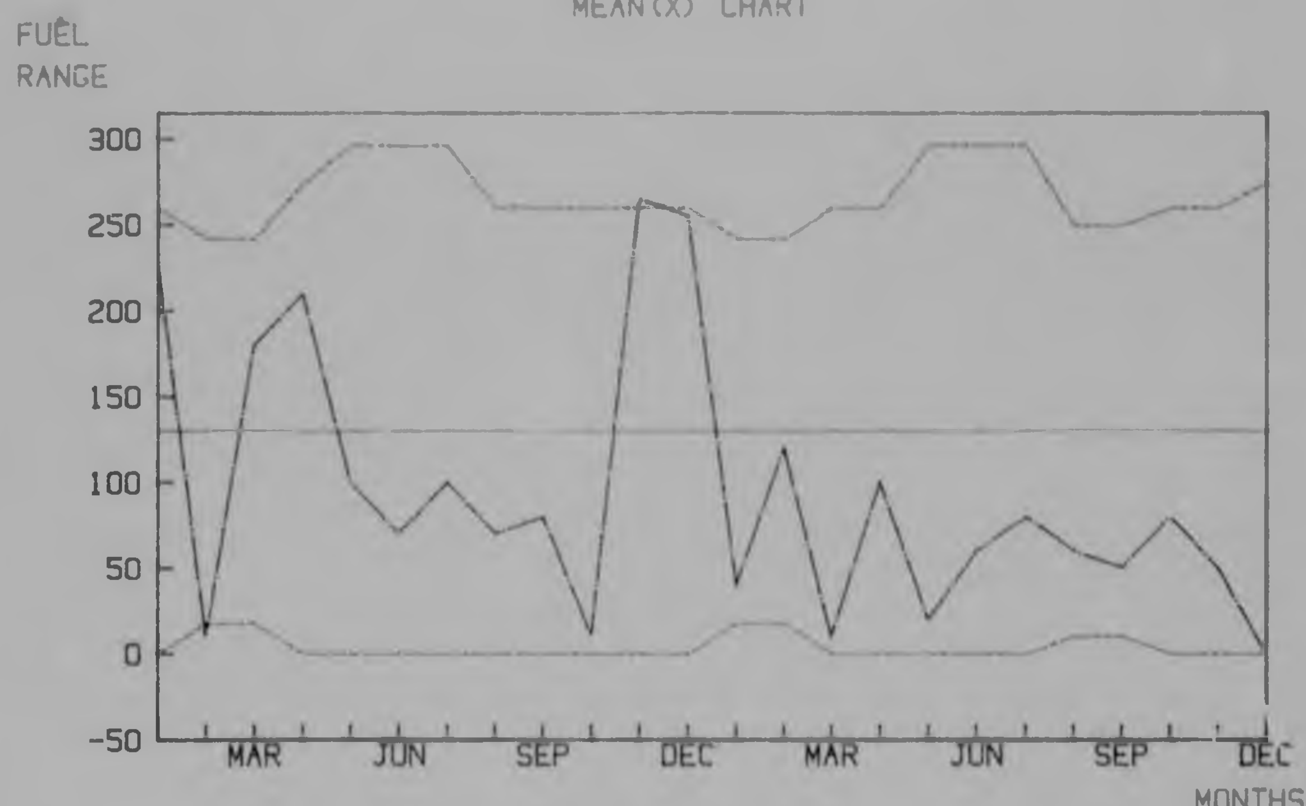
BRANCH: CLEANSING DEC 1983

DEPOT : SELBY (PETROL)

TYPES ALL



MEAN (X) CHART



RANGE (R) CHART

1982 - 1983

FIG 7.15.3.2

CONTROL CHART (3 SIGMA)

BRANCH: CLEANSING DEC 1983

DEPOT . ALL (PETROL)

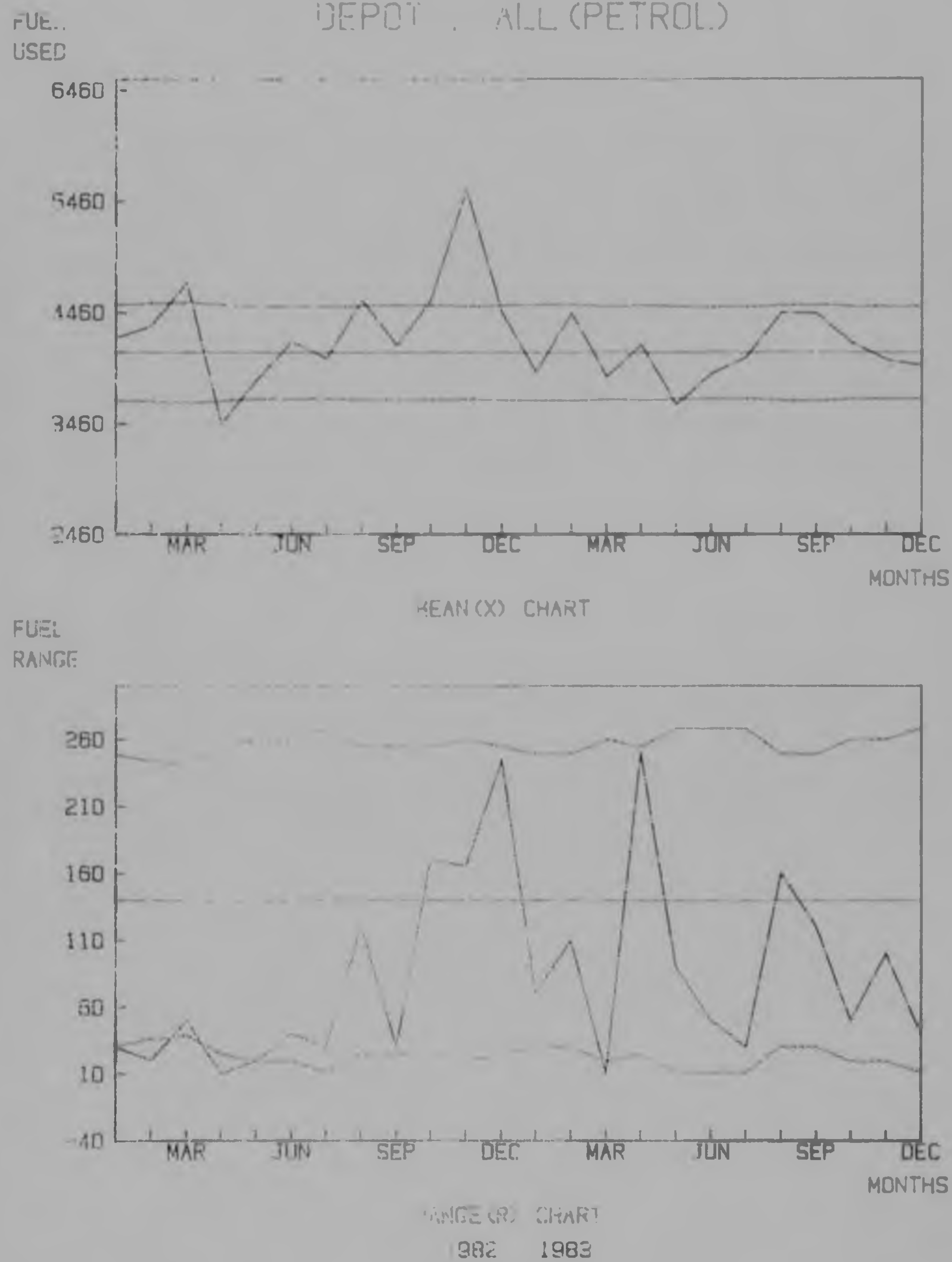
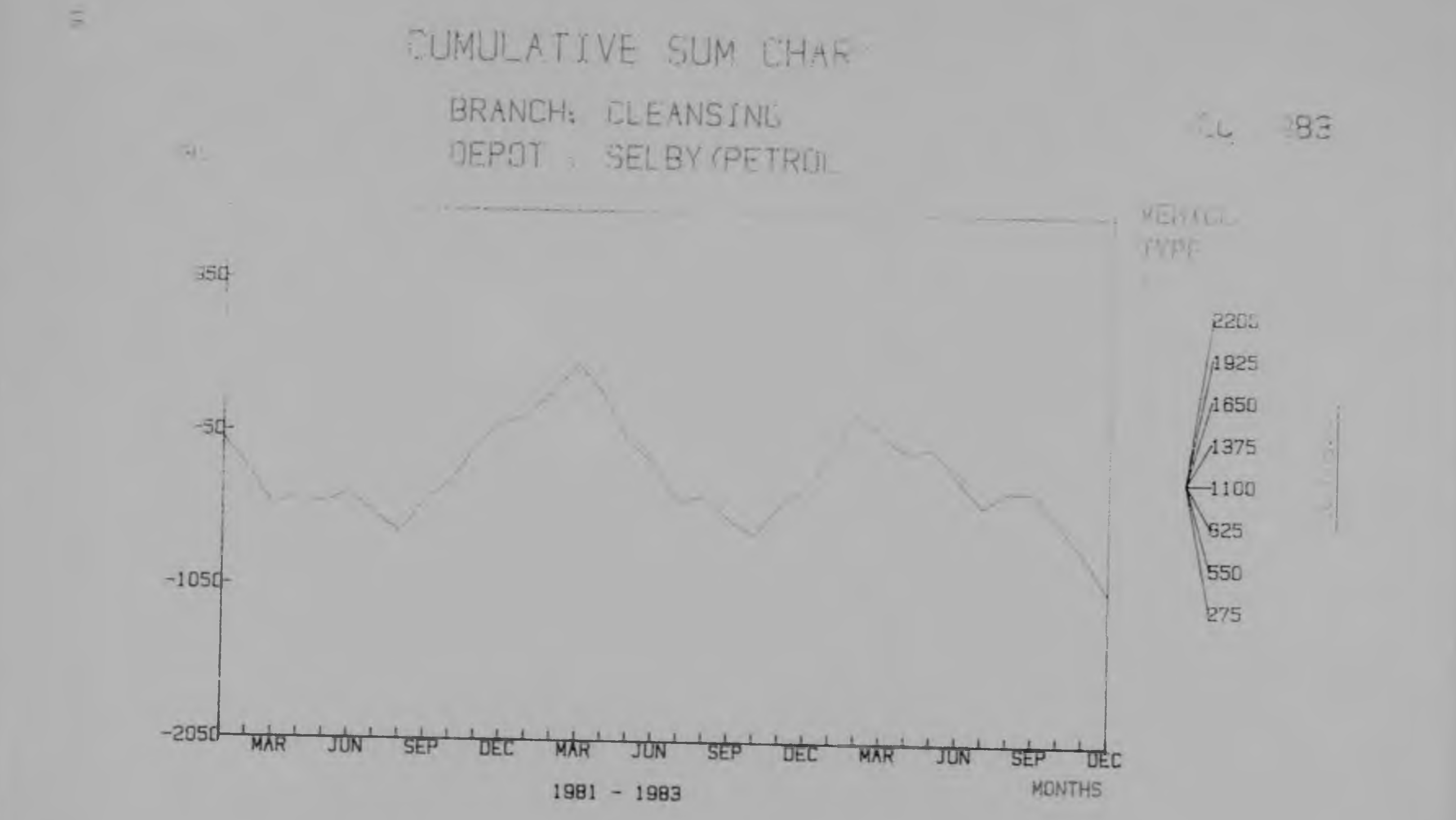


FIG 7.15.3.3


CUMULATIVE SUM CHAR
BRANCH: CLEANSING
DEPOT : SELBY (PETROL
83
350
-50
-1050
-2050
MAR
JUN
SEP
DEC
MAR
JUN
SEP
DEC
MAR
JUN
SEP
DEC
1981 - 1983
MONTHS
TYPE
2200
1925
1650
1375
1100
825
550
275

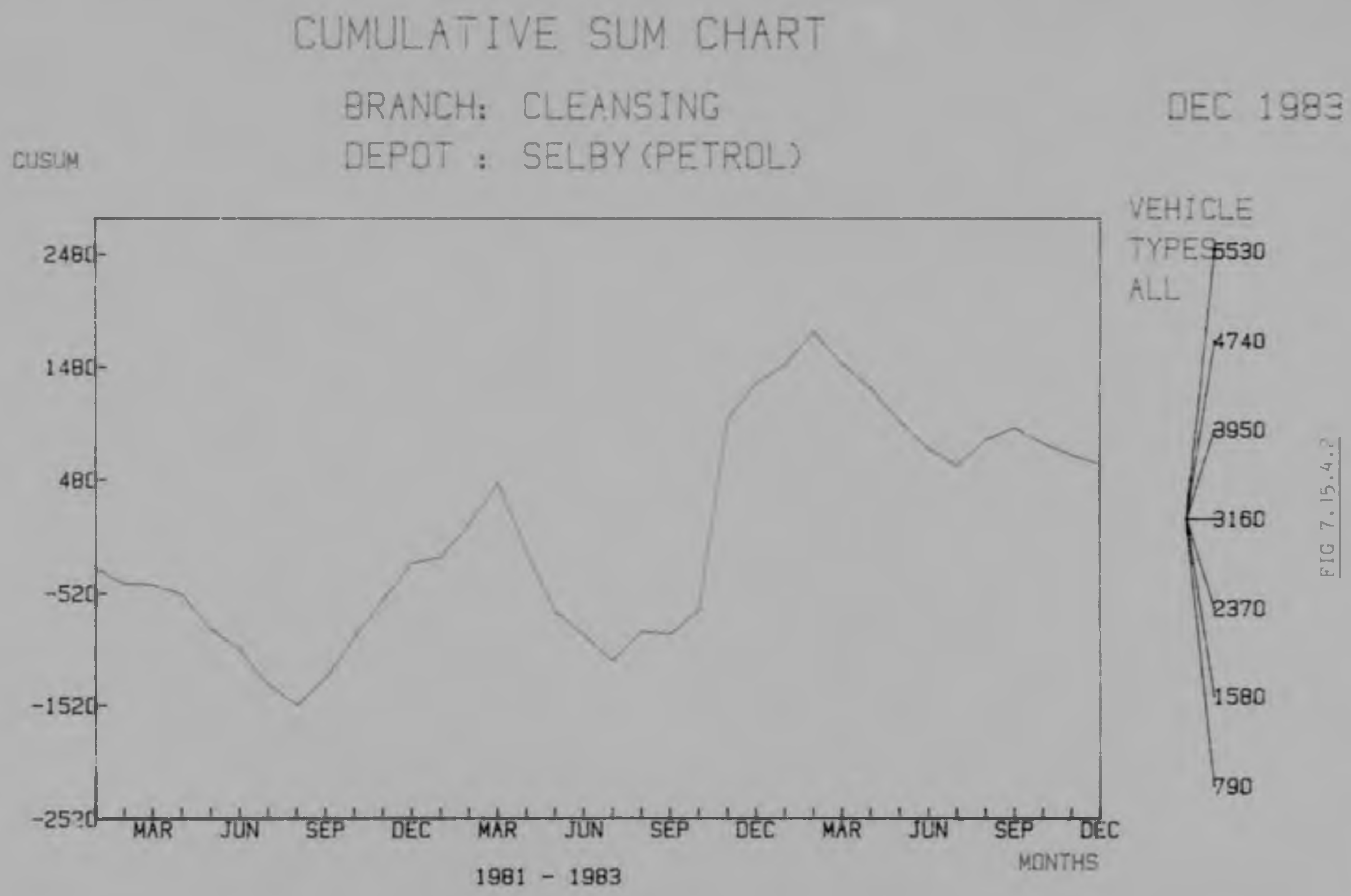

CUMULATIVE SUM CHART
BRANCH: CLEANSING
DEPOT : SELBY (PETROL)
DEC 1983
CUSUM
2480
1480
480
-520
-1520
-2520
MAR
JUN
SEP
DEC
MAR
JUN
SEP
DEC
MAR
JUN
SEP
DEC
MONTHS
1981 - 1983
VEHICLE
TYPES
ALL
5530
4740
3950
3160
2370
1580
790

FIG 7.15.4.2

CUMULATIVE SUM CHART

BRANCH: CLEANSING

DEPOT : ALL(PETROL)

DEC 1983

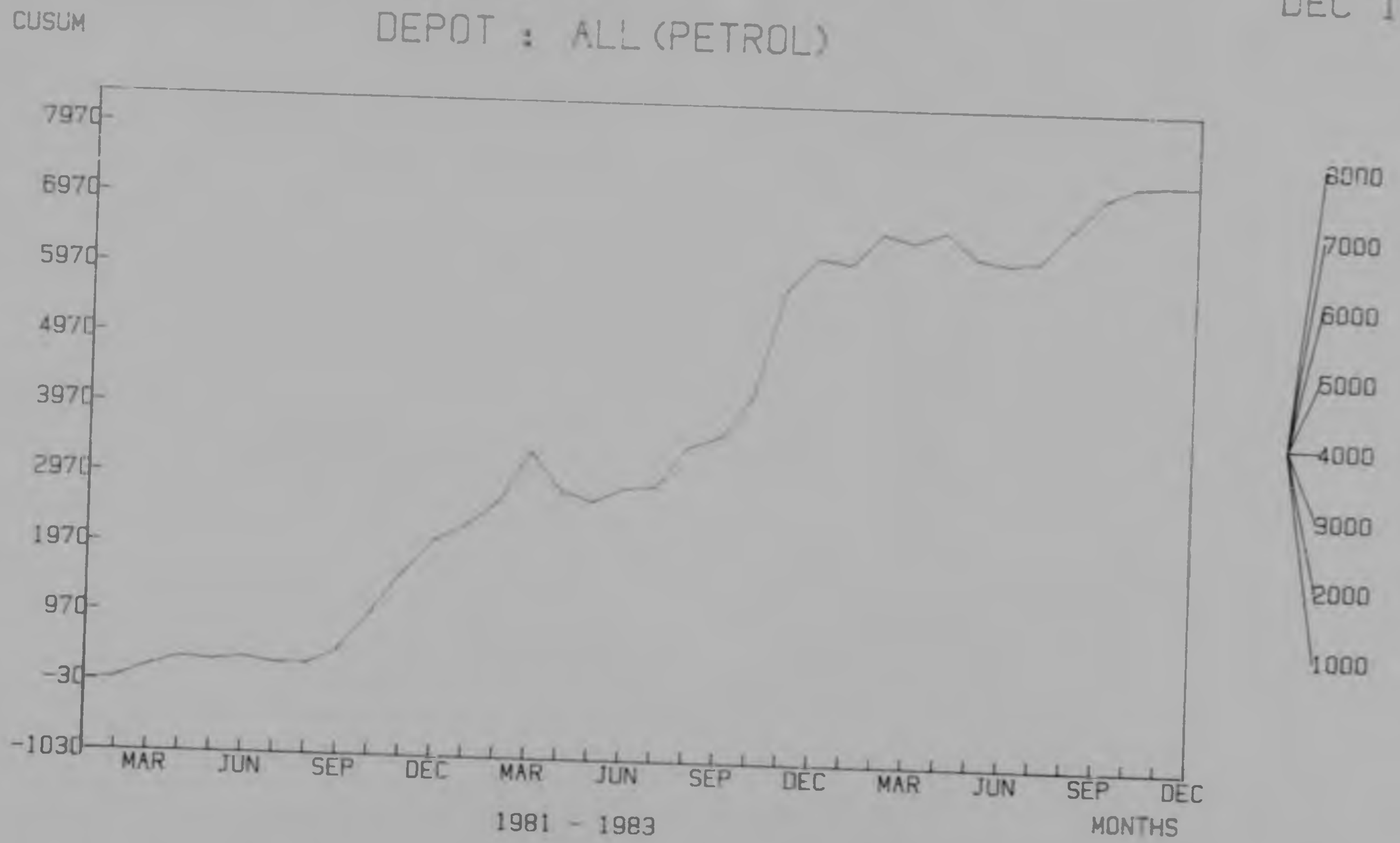


FIG 7.15.4.3

FUEL CONSUMPTION FIGURES AT DEC 1983
BRANCH: CLEANSING
DEPOT : SELBY
TYPE: 1
FUEL: PETROL

PAGENO: 1

MONTH/YEAR	THIS MONTH	YEAR TO DATE
JAN 1981	1000	1000
FEB 1981	900	1900
MAR 1981	860	2760
APR 1981	1170	3930
MAY 1981	1050	4980
JUN 1981	1180	6160
JUL 1981	950	7110
AUG 1981	990	8100
SEP 1981	1310	9410
OCT 1981	1220	10630
NOV 1981	1350	11980
DEC 1981	1240	13220
JAN 1982	1150	14370
FEB 1982	1260	1260
MAR 1982	1280	2540
APR 1982	900	3440
MAY 1982	800	4240
JUN 1982	950	5190
JUL 1982	850	6040
AUG 1982	1150	7190
SEP 1982	940	8130
OCT 1982	1000	9130
NOV 1982	1310	10440
DEC 1982	1210	11650
JAN 1983	1320	12970
FEB 1983	1380	1380
MAR 1983	940	2320
APR 1983	990	3310
MAY 1983	1130	4440
JUN 1983	900	5340
JUL 1983	920	6260
AUG 1983	1200	7460
SEP 1983	1100	8560
OCT 1983	900	9460
NOV 1983	900	10360
DEC 1983	850	11210

FIG 7.15.5

TABLE OF BRANCHES AND DESCRIPTIONS
==================================

DATE: DEC 1983 PAGENO: 1

CODE	DESCRIPTION
1	CLEANSING
2	NONE
3	NONE
4	NONE
5	NONE
6	NONE
7	NONE
8	NONE
9	NONE
10	NONE
11	NONE
12	NONE
13	NONE
14	NONE
15	NONE
16	NONE
17	NONE
18	NONE
19	NONE
20	NONE
21	NONE
22	NONE
23	NONE
24	NONE
25	NONE
26	NONE
27	NONE
28	NONE
29	NONE
30	NONE
31	NONE
32	NONE
33	NONE
34	NONE
35	NONE
36	NONE
37	NONE
38	NONE
39	NONE
40	NONE
41	NONE
42	NONE
43	NONE
44	NONE
45	NONE
46	NONE
47	NONE
48	NONE
49	NONE
50	NONE

PROCESSING COMPLETE

*** LISTING COMPLETE ***

FIG 7.15.6.1

TABLE OF DEPOTS AND DESCRIPTIONS

DATE: DEC 1983 PAGENO: 1

CODE	DESCRIPTION
1	SELBY
2	NORWOOD
3	SOUTHDALE
4	NONE
5	NONE
6	NONE
7	NONE
8	NONE
9	NONE
10	NONE
11	NONE
12	NONE
13	NONE
14	NONE
15	NONE
16	NONE
17	NONE
18	NONE
19	NONE
20	NONE
21	NONE
22	NONE
23	NONE
24	NONE
25	NONE
26	NONE
27	NONE
28	NONE
29	NONE
30	NONE
31	NONE
32	NONE
33	NONE
34	NONE
35	NONE
36	NONE
37	NONE
38	NONE
39	NONE
40	NONE
41	NONE
42	NONE
43	NONE
44	NONE
45	NONE
46	NONE
47	NONE
48	NONE
49	NONE
50	NONE
51	NONE
52	NONE
53	NONE
54	NONE

FIG 7.15.6.2 (1)

TABLE OF DEPOTS AND DESCRIPTIONS

DATE: DEC 1983 PAGENO: 2

CODE	DESCRIPTION
55	NONE
56	NONE
57	NONE
58	NONE
59	NONE
60	NONE
61	NONE
62	NONE
63	NONE
64	NONE
65	NONE
66	NONE
67	NONE
68	NONE
69	NONE
70	NONE
71	NONE
72	NONE
73	NONE
74	NONE
75	NONE
76	NONE
77	NONE
78	NONE
79	NONE
80	NONE
81	NONE
82	NONE
83	NONE
84	NONE
85	NONE
86	NONE
87	NONE
88	NONE
89	NONE
90	NONE
91	NONE
92	NONE
93	NONE
94	NONE
95	NONE
96	NONE
97	NONE
98	NONE
99	NONE

*** LISTING COMPLETE ***

FIG 7.15.6.2 (2)

8. CONCLUSIONS

8.1. System Investigation

Although numerous irregularities were uncovered by this investigation, some of which are easily rectifiable, many of these are remnants from outdated methods of information gathering and would require extensive changes to the philosophies incorporated in the system. Examples of this are the month-end fill up procedure and separation of duties and localisation of accountability as discussed in Section 5. Such a revamp is costly in terms of both time and manpower and is not recommended lightly. However, it should be clearly understood that the accuracy of any data processing system has as a basic limitation, the accuracy of the input data, and consequently to invest large sums of money to process inaccurate data is somewhat foolhardy.

It should be noted that this investigation was carried out only on the Cleansing Branch of the C.E.D., the philosophy being that any irregularities found here are probably generalisable to the remaining branches. This, of course, may not be true and a similar investigation would have to be carried out in these branches if certainty were required prior to making some commitment.

It was felt at the time that the approach of information gatherer was more likely to earn the co-operation of the operations personnel concerned, than that of inspector and this, in fact, proved to be invaluable, laying the emphasis on the desire to make the system more efficient rather than more watertight.

In general, it is believed that a good evaluation of the existing system in the Cleansing Branch has been achieved and that it is probably capable of generalisation to the other branches of the Department.

8.2 Computer System Design and Implementation

The final graphs from the system show that the concept of utilising statistical control techniques as a tool for management of fuel consumption is a good one. Certainly within the given constraints, the system provides data which may be used effectively to determine those uncontrollable variations caused by chance from those which have determinable causes.

Some practical difficulties arose which made the utilisation of the existing processor unsuitable for generalised use of this package. These are:

8.2.1 Insufficient diskette space for the amount of data required to be stored. This could be obviated with the upgrading of the machine to contain more disc memory.

8.2.2 Time required to load data into the system. This is not a serious problem while only the Cleansing Branch is using this system but manual loading of all data from all branches would take far too long. This could be solved by creating a diskette from the existing fleetmaster system and using this as the input to the system. This of course introduces the problem of inaccurate data discussed above, and some corrective measures would need to be taken in conjunction with the introduction of the utilisation of such data.

Another difficulty is that generalised

acceptance of the system appears to be a major stumbling block. In the case of management, the resistence to change factor is high since introduction of these techniques would mean the necessity of becoming familiar with somewhat unfamiliar statistical theory and in the case of shop floor personnel, the belief that the system will "spy" on them and report their activities, results in a great unwillingness to co-operate in making the system work.

8.3 General

Possibly the major achievement of this project has been to irrefutably show to the management of the City Engineers Department of Johannesburg the need to perform a major re-evaluation of their fuel usage gathering and recording process and the definite advantages of the concept of utilising statistical quality control techniques as a tool to control that fuel usage.

9. APPENDICES

A list of all appendices is given below:

APPENDIX	DESCRIPTION
A	Background in Statistical Theory
B	Examples of CED documents
C	Example of Possible Error in Fuel Computation
D	Example of Commercial Fuel Management System
E	System Data File Characteristics
F	Discussion on Range Calculations
G	Change in Usage is slope of Cusum
H	Detailed System Breakdown

APPENDIX

Determination of Standard Deviatio[illegible]

Grant (1980) defines the standard deviation or root-mean-square deviation about the arithmetic mean as:

$$\sqrt{\frac{(X_1 - \overline{X})^2 + (X_2 - \overline{X})^2 + \text{----} (X_N - \overline{X})^2}{N}}$$

where X_1 , X_2 ______ X_N are occurences of X

$\overline{X}$ is the arithmetic mean

N is the no of occurences

and furthermore defines [illegible] relationship between the average range [illegible] and the [illegible] of the universe [illegible]

[illegible] subgroup sizes, N.

From Grant (1980) it can be shown that the standard deviation of a sample, $\sigma_{\overline{X}}$ is related to the standard [illegible] universe by:

[illegible]

[illegible] operator

$$
\begin{aligned}
VAR(\overline{X}) &= \tfrac{1}{N^2} \cdot VAR(\textstyle\sum X \text{[illegible]} \\
&= \tfrac{1}{N^2} \textstyle\sum VAR(X \text{[illegible]} \\
&= \tfrac{1}{N^2} (N\sigma^2) \\
&= \frac{\sigma^2}{N}
\end{aligned}
$$

and $\sigma_{\overline{X}} = \dfrac{\sigma}{\sqrt{N}}$

A.2 Central Limit Theory

Braverman (1981) describes a simple test for testing the assumption that observed data came from a normally distributed population. This is achieved by plotting sample data on normal probability graph paper. On such paper a normal distribution will plot as a straight line. If this test is inconclusive a chi square goodness of fit test may be performed. In most cases though, the assumption of normalcy is acceptable unless there are factors to indicate otherwise.

Furthermore, the theory of the Central Limit Theorem is applicable. This Theorem states, in the words of Grant (1980):

"Irrespective of the slope of the distribution of a universe, the distribution of average values, $\bar{X}$'s, of subgroup of size n, ($\bar{X}_1$, $\bar{X}_2$ ---- $\bar{X}k$), drawn from that universe will tend towards a normal distribution as the subgroups size, n, grows without bound".

He also shows that n need not be very large before the normal distribution may be applied. Subgroups of the size used in each vehicle type fit into this category, and the assumption of normalcy will be made.

A.3 Subgroup Sizes

One of the most important factors in the setting up of control charts is the size of the sample (subgroup size). Here there are various factors to be considered.

The smaller the subgroup, the greater the likelihood that any assignable causes which might occur will show up as variations between subgroups rather than in differences between members of a subgroup (Duncan 1974). Furthermore, where the cost of measurement is high, large subgroups may be prohibitive.

However, if subgroups are taken from a non-normal universe, they should not be made too small (See Appendix A.2). Also the accuracy with which the range, $\bar{R}$, may be used in place of the standard deviation diminishes with increasing subgroup size.

In the present case, subgroup sizes are largely predetermined by the number of existing vehicles in each type, and thus the above discussion is largely academic. Of importance is the fact that in this case, subgroup sizes change from month to month, and since the control limits are based on subgroup size, these too must change accordingly.

A.4 Theory of Extreme Runs

Making the assumption that the probability that any measurement (x) will fall above the universe mean, is $\frac{1}{2}$ i.e. assuming a symmetrical distribution we have

$$P(x) = \frac{1}{2}$$

and the probability of two successive measurements is

$$P(x).P(x) = \frac{1}{4}$$

Likewise, the probability that 7 successive process measurements will fall either all above or all below this mean is:

$$P(x)^7 + P(x)^7 = (\tfrac{1}{2})^7 + (\tfrac{1}{2})^7 = \frac{1}{64}$$

It is thus possible to determine the probability of a run being a chance variation and thus to ascertain the significance of such a run -

Grant (1980, considers the following as significant:

- Seven successive points on the same side of the mean
- Ten out of 11 successive points on the same side of the mean
- Twelve out of 14 successive points on the same side of the mean
- Fourteen out of 17 successive points on the same side of the mean
- Sixteen out of 20 successive points on the same side of the mean

These may be used in conjunction with 3σ limits to determine assignable causes.

A.5 Abridged Table of Factors for Determining 3σ Control Limits From R for $\overline{X}$ and R Charts

An extract from the table of factors for determination of 3σ control limits is reproduced from Grant on the following page in Table A.5.1.

Grant (1980) considers the following as significant:

- Seven successive points on the same side of the mean
- Ten out of 11 successive points on the same side of the mean
- Twelve out of 14 successive points on the same side of the mean
- Fourteen out of 17 successive points on the same side of the mean
- Sixteen out of 20 successive points on the same side of the mean

These may be used in conjunction with 3σ limits to determine assignable causes.

A.5 Abridged Table of Factors for Determining 3σ Control Limits From R for $\bar{X}$ and R Charts

An extract from the table of factors for determination of 3σ control limits is reproduced from Grant on the following page in Table A.5.1.

TABLE A.5.1			
Size of Subgroup	$\overline{X}$ Chart Factor	$\overline{R}$ Chart Factors	
		Lower Limit	Upper Limit
n	A2	D3	D4
2	1,88	0	3,27
3	1,02	0	2,57
4	0,73	0	2,28
5	0,58	0	2,11
6	0,48	0	2,00
7	0,42	0,08	1,92
8	0,37	0,14	1,86
9	0,34	0,18	1,82
10	0,31	0,22	1,78
11	0,29	0,26	1,74
12	0,27	0,28	1,72
13	0,25	0,31	1,69
14	0,24	0,33	1,67
15	0,22	0,35	1,65
16	0,21	0,36	1,64
17	0,20	0,38	1,62
18	0,19	0,39	1,61
19	0,19	0,40	1,60
20	0,18	0,41	1,59

APPENDIX B

B.1 Example of Fuel Issue Coupon:

These coupons are issued to the depots for use in recording all issues of fuel. The books are printed in 5, 10, 20 and 30 litre denominations. An example of a 20 litre petrol coupon is shown in fig. B.1.

B.2 Fuel Reconciliation Statement

This statement is used by each depot to match:

- The pump readings at start and end of day to the issues for the day.
- The dip readings at start and end of day to the issues and receipts.
- The intertank transfers to the dip tank readings at start and end of day.

An example of a reconciliation statement is shown in fig. B.2.

B No. 121101

STADSINGENIEURSAFDELING
CITY ENGINEER'S DEPARTMENT

PETROL

HIGH RATE* HOË GRAAD*
LOW RATE LAE GRAAD

TEENBLAD
COUNTERFOIL

NIE GELDIG VIR UITREIKING NIE
NOT VALID FOR ISSUE

DATUMSTEMPEL
DATE STAMP

20

LITER
LITRES

VOERTUIG No.
VEHICLE No.

TJ

KOEPON UITGEREIK DEUR:
COUPON ISSUED BY:

*SKRAP WAT NIE VAN TOEPASSING IS NIE.
DELETE INAPPLICABLE WORD.

SS/CS 101/A (CED) M.81

B No. 121101

STADSINGENIEURSAFDELING
CITY ENGINEER'S DEPARTMENT

PETROL

HIGH RATE* HOË GRAAD*
LOW RATE LAE GRAAD

LET WEL: KOEPON SLEGS GELDIG OP UITREIKDATUM.
NOTE: COUPON VALID ONLY ON DATE OF ISSUE.

Kol./Col.

KAARTTIPE
CARD TYPE 1–3

AFDELING
DEPT. 4–6

DAG DAY MAAND MONTH JAAR YEAR

DATUM
DATE 7–12

VOERTUIG No.
VEHICLE No. 13–16

TJ No. TJ 17–22

DATUMSTEMPEL
DATE STAMP

20

LITER
LITRES

MOET DEUR POMPBEDIENER INGEVUL WORD:
TO BE FILLED IN BY PUMP ATTENDANT:

AFSTANDMETERLESING (m/km/h)*
SPEEDO READING (m/km/u)*

POMPLESING NA INTAP (LITER)
PUMP FINISH (LITRES)

POMPLESING VOOR INTAP (LITER)
PUMP START (LITRES)

NETTO HOEVEELHEID UITGEREIK (LITER)
NETT ISSUED (LITRES) 2 0 0 0 44–47

POMPKODE
PUMP CODE 74–76

UITGEREIK DEUR POMPBEDIENER
ISSUED BY PUMP ATTENDANT ....

ONTVANG DEUR
RECEIVED BY ....

NAAM IN BLOKLETTERS
NAME IN BLOCK LETTERS

M.81

FIG B.1

CITY ENGINEER'S DEPARTMENT

RECONCILIATION STATEMENT

PETROL/DIESELINE FUEL TANK

TO : MECHANICAL WORKSHOPS

PUMP NO | 1 | | 3 |

	CURRENT
PUMP READING AT START	4 … 9
ISSUES	10 … 13
PUMP READING AT END	14 … 19
DIP READING AT START	20 … 24
RECEIPT (D/N ATTACHED)	25 … 29
DIP READING AT END	30 … 34
INTERTANK TRANSFER	35 … 38
FROM	39 … 41
TO	42 … 44
DELIVERY NOTE NO.	45 … 52
TANK CAPACITY	53 … 57
REFERENCE NUMBER 75	76 … 80

SS/CS 1023/B

M 98

DISTRICT ENGINEER

FIG B.2

APPENDIX C

C.1 Example of possible error in fuel consumption estimates:

The method of recording fuel usage determines that only the fuel actually dispensed into a vehicle during a given month is used to compute the consumption figures over a distance measured from the first odometer reading to the last one in that month with no fill-up at month end. A worst case analysis is done below using typical figures to estimate possible inaccuracies.

C.1.1 Correct Analysis

TABLE C.1.1

Month	Date	Kilometers Travelled	Actually Used	Quantity Dispensed	In Tank After Fill-up
1	31/03	2000	-	50	100
2	05/04	2500	50	20	70
	20/04	3000	50	30	50
	30/04	3500	50	100	100
Total		1500	150	150	

$$\text{Now actual consumption} = \frac{A}{B} = \frac{1500}{150} = 10{,}0 \text{ km}/\ell$$

$$\text{Recorded consumption} = \frac{A}{C} = \frac{1500}{150} = 10{,}0 \text{ km}/\ell$$

C.1.2 Erroneous Analysis

TABLE C.1.2					
Month	Date	Kilometers Travelled	Actually Used	Quantity Dispensed	In Tank After Fill-up
1	31/03	2000	-	0	50
2	05/04	2500	50	100	100
	20/04	3000	50	40	90
	30/04	3500	50	60	100
Total		1500	150	200	

$$\text{Actual consumption} = \frac{A}{B} = \frac{1500}{150} = 10{,}0 \text{ km}/\ell$$

$$\text{Recorded consumption} = \frac{A}{C} = \frac{1500}{200} = 7{,}5 \text{ km}/\ell$$

$$\text{Percentage Error} = \frac{10-7{,}5}{10} \times 100 = 25\%$$

As can be seen, if there is no requirement to fill the vehicle at month end, there is no method of verifying during which month fuel dispensed into the vehicle is used. In the long run of course, these fluctuations randomise out but any figures based on monthly consumption data or comparisons to previous years monthly data may be erroneous by up to 25% and decisions based on such information could be invalid.

APPENDIX D

A commercially available system called Vanguard II, utilises on board microcomputers to provide management information on fuel usage, driver performance, vehicle performance, vehicle utilisation, trip reports and vehicle service schedule. Examples of some of these schedules are shown below:

D.1 Fuel Usage Report

See fig. D.1

D.2 Vehicle Performance Report

See fig. D.2

D.3 Driver Performance Report

See fig. D.3

D.4 Trip Report

See fig. D.4

VANGUARD FUEL USAGE REPORT

Page 1

Report date : 01/10/82

Depot : BENONI   Group : MERC. RIGID   Period : 01/09/82 to 30/09/82

Vehicle No.	Distance(KMS)	Fuel Used (LTR)	Ltr/100km	12Mth Average	Eoge Variance	Total Revs (Mill)	Revs/Ltr
01234	5670	2019	35.6	34.2	4,14	11.34	5616
05759	7696	2470	32.1	35.0	8.3	13.32	5393
02835	4570	1808	39.5	30.6	29.14	7.81	5426
06169	4235	1406	33.2	31.5	5.44	10.89	7745
oup Total :	22179	7703	34.7	32.8	5.84	42.36	5889

FIG D.1

VANGUARD VEHICLE PERFORMANCE REPORT

Page 1.

Report Date : 01/10.82

Depot : BENONI   Group : MERC. RIGID   Period : 01/09/82 to 30/09/82

Vehicle No.	Driver No.	Odometer Reading	Distance (kms)	Engine Hours	Road Hours	Engine Idle	Driving Efficiency	% Time over Opt. RPM	Occur over Max. RPM	Time over Speed Lim.	Ltr/100Km
01234	2215	55195	2156	51.05	47.55	3.10	76,0	15,3	7	9.05	35,6
	2218		3514	75.00	72.30	2.30	82.1	9,8	1	7.53	
05759	2213	128591	3015	62.58	59.38	3.20	78,0	18,3	5	10.17	32,1
	2211		4681	85.00	80.35	4.25	79,0	16,1	10	11.01	
02835	2216	115216	4578	108.59	97.28	11.31	83,5	11,2	9	9.48	39,5
06169	2212	261553	1098	35.10	33.58	1.12	83,0	8,4	4	3.49	33,2
	2219		3137	86.05	80.53	5.12	83,0	9,9	6	10.55	[illegible]
Group Totals			22179	504.17	472.57	31.20	80,9	12,7	42	53.48	34,7

FIG D.2

VANGUARD DRIVER PERFORMANCE REPORT

Report Date : 01/10/82 Page 1

Depot : BENONI Period : 01/09/82 to 30/09/82

Driver No.	Standard Time	Total Time	Operating Time Actual	Operating Time %age	Driving Time	Engine Idle Time	% Driver Efficiency	% Over Opt RPM	Occur Over Max RPM	Distance (KMS)	Max Speed	% Time Over Speed Limit
2215	152.00	160.45	75.02	46.2	51.05	3.10	76.2	[illegible]	7	2156	107.9	17.80
221[illegible]	152.00	144.05	102.03	70.8	75.00	2.30	82.0	9.8	1	3514	113.2	10.51
221[illegible]	152.00	106.50	88.35	82.9	62.58	3.20	70.1	15.8	5	3015	[illegible]	[illegible]
221[illegible]	152.00	151.54	110.00	72.4	85.00	4.25	79.4	12.9	[illegible]	4681	103.5	12.96
221[illegible]	168.00	160.01	134.58	84.3	108.59	[illegible]	81.2	9.5	9	4578	100.5	14.02
221[illegible]	96.00	92.30	49.16	53.3	35.10	1.12	85.6	9.3	4	1098	109.6	15.73
2217	152.00	140.14	100.10	71.4	86.04	5.12	[illegible]	11.5	6	3137	111.1	12.75

FIG D.3

VANGUARD TRIP REPORT

Reports numbered sequentially per vehicle for ease of reference and security

Report No. 005 for Vehicle No. 2983 — User set vehicle No.

Period : Thurs. 9th Sept. 1982 - 18.02 — Date and time of previous report
Sat. 11th Sept. 1982 - 06.36

Odometer : 025461 — Current odometer reading

Driver		Depart Time	Arrive Time	Distance (Kms)	Road Hours	Engine Idle	Stop Hours
123456	10/09	SIGN-ON	08.00			00.15	00.30
		08.30	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
		09.10	09.30	5.0	00.20	00.03	01.00
		10.30	14.30	200.0	04.00	00.10	00.30
		15.00	19.30	215.0	04.30	00.0[illegible]	0[illegible]
		19.35	SIGN-OFF				
				430.0	09.20	00.33	02.1[illegible]
199999	10/09	SIGN-ON	20.04			00.14	00.24
		20.28	23.00	140.0	02.32	00.10	05.00
	11/09	04.00	06.10	138.9	02.10	00.01	00.25
		06.35	SIGN-OFF				
				278.9	04.42	00.25	05.49

Driver inserted his unique key into Vanguard II at 8.00 a.m. — all activity up to sign-off at 19.35 clearly recorded

Records depart time from depot after user set distance exceeded

Excessive idling noted.

Stop recorded after user set parameter exceeded.

Driver	123456	99999	Total
Total Distance (Kms)	430.0	278.9	708.9
Engine Hours	09.53	05.07	15.00
Engine Idle Time	00.33	00.25	00.58
Road Hours	09.20	04.42	14.02
Stop Hours	02.15	05.49	08.04
Time Below 1250 RPM	00.57	00.25	01.22
Time Above 1750 RPM	01.09	00.50	01.59
Within Optimum RPM	06.50	03.15	10.05
Driving Efficiency %	76.5	72.2	75.1
Occurance Over 2250 RPM	3	2	5
Maximum RPM Achieved	2360	2450	2450
Exceed 80KPH - Occurances	5	6	11
Exceed 80KPH - Time	01.25	01.40	03.05
Maximum Speed Achieved	106.7	115.0	115.0
Free Wheel - Occurances	1		1
Free Wheel - Time	00.02		00.02
Fuel Loaded (Litres)			259.5
Consumption (Ltrs/100kms)			36.6

User set R.P.M. bands and speed limits ensure tight management control over driving habits. Vehicle abuse or dangerous driving is highlighted immediately for prompt action.

IMMEDIATE FUEL CONSUMPTION FACTOR

Immediate reporting of free wheeling occurance enables instant action by driver supervisor

Individual driver performance clearly itemised

▨ Indicates user set parameters

FIG D.4

APPENDIX E

E.1 Index File Simulation using Two Relative Files:

Two Relative files have been created for each branch-depot combination. For example, the files for branch No. 2, Depot No. 1, might look as follows:

INDEX FILE A

← TYPE 1	x		TYPE 2	x		TYPE 3 →				← TYPE 99		→
Position No 2	Default List	Fuel Type	Position No ƀ	Default List	Fuel Type	Position No 1	Default List	Fuel Type		Position No ƀ	Default List	Fuel Type

SEQUENTIAL FILE B

← TYPE 3				x	TYPE 1 →					
Ind = 1	36 x	Monthly Total;	No of Vehicles;	Vehicle Range;	Ind = 1	36 x	Monthly Total;	No of Vehicles;	Vehicle Range;	Ind = ƀ

- The relative record number in file A bears a direct relationship to the vehicle type i.e. vehicle type 3 is relative record 3.

- The position number on file A determines the relative number of the record in which the data for that type may be found e.g. the data of type 3 (relative record 3 on file A) may be found in file B at relative record No. 1.

- The default list and fuel type are short fields and do not substantially increase the size of the file. Hence their inclusion in the tag file, A.

- The indicator of file B indicates whether that record has been used or not e.g. if we wanted to load data for type 4, the program would look for the first blank record (not used), write the data onto file B and the current position onto file A and change the indicator to 1. Record 3 on file B would now be considered "used".

This feature makes it possible to access these files sequentially, i.e. in the order they were loaded, or directly i.e. only a specific record or a series of records in the order of loading, thus overcoming the difficulty of the lack of indexed access on this micro.

There is a penalty to be paid in the form of a drop in processing time. However, since most of these jobs are running at operator speed this difference in time is inconsequential.

E.2 File Space Calculations:

E.2.1 File A - Index File:

No. of bytes for:	Position No.	=	8
	Default List	=	6
	Fuel Type	=	4
	Total/Type	=	18

No. of vehicle types in a depot	=	99
Therefore : Total/Depot	=	1782
Now 7 blocks of 256 bytes	=	1792
Therefore : Total per file	=	1792

E.2.2 File B - Sequential Data File:

No. of bytes for:	Indicator	=	8
	Total Fuel	=	8
	No. of Vehicles	=	8
	Range	=	8
	Total	=	32

and for 36 months Total/Type	=	1152

Now on average there are only 20 vehicle types used in a depot, therefore using 40 for expansions:

Total per depot	=	46080
Now 180 blocks of 256 bytes	=	46080
Total per file	=	46080

E.2.3 File C - Sequential Parameter File:

No. of bytes for:	Position No.	=	8
	Historical Mean	=	8
	Historical Range	=	8
	Total	=	24

and maximum probable No. of vehicle types per depot	=	40
Total/Depot	=	960
Now 4 blocks at 256 bytes	=	1024
Total per file	=	1024

The average No. of depots in a branch is 4.

Total per depot for all files	=	48896
and assuming 5 depots/branch, then		
Total per Branch	=	244480 characters.

Each diskette is capable of holding 1 000 000 characters.

It would therefore seem expedient to load four branches onto each diskette, although in the event that the expansion is greater than envisaged, it is possible that only three or two will be feasible.

Required by the system:

Master diskette	(programs and master files)	- 1 diskette
Data diskettes	(16 branches)	- 4 diskettes
Total		- 5 diskettes

APPENDIX F

F.1 Calculation of Moving Average

Assuming a set of occurrences in months 1 to 12 of $a_1, a_2, \ldots\ldots a_{12}$ then the moving average is given by:

$$\frac{a_1 + a_2 + a_3 + a_4 \ldots\ldots\ldots\ldots a_{12}}{12} = \frac{\sum_{i=1}^{12} a_i}{12}$$

Now this average should be applied to the midpoint of the moving average period i.e. months 6 or 7. However, in practice the average only becomes available at the end of the twelfth month and it is both simpler in calculation as well as psychologically to apply the average to the twelfth month.

The moving average for the following month would be given by:

$$\frac{a_2 + a_3 + a_4 + a_5 \ldots\ldots\ldots\ldots a_{12} + a_{13}}{12}$$

This may also be obtained by:

$$\frac{\sum_{i=1}^{12} a_i + a_{13} - a_1}{12}$$

and generalising:

$$MA_i = \frac{MT_{i-1} - T_{i-n} + T_i}{n}$$

where n = moving average spread
MT = moving total
MA = moving average
T = occurrence total

F.2 Calculation of Moving Range

Assuming a set of occurrences in months 1 to 12 arranged in ascending order of magnitude as a_1, a_2 .... a_{12} then the moving range is given by:

$a_{12}-a_1$

As in the case of the moving average this should be applied to the midpoint of the moving range period. However, in practice it is applied to the twelfth month. The thirteenth month's moving range would be given by:

$a_{13}-a_2$

where a_2, a_3 .... a_{13} are the 12 occurrences in ascending order of magnitude, and generalising

MR = T max - T min

where MR = moving range
T = occurrence total

F.3 Range for Branch and Depot Charts

An extension to the control chart technique utilising moving averages (Grant 1980) and moving ranges overcomes the difficulty of comparing ranges of dissimilar vehicle types.

The problem is illustrated below:

Assume we have two vehicle types, with widely varying fuel usages

F.2 Calculation of Moving Range

Assuming a set of occurrences in months 1 to 12 arranged in ascending order of magnitude as a_1, a_2 .... a_{12} then the moving range is given by:

$a_{12}-a_1$

As in the case of the moving average this should be applied to the midpoint of the moving range period. However, in practice it is applied to the twelfth month. The thirteenth month's moving range would be given by:

$a_{13}-a_2$

where a_2, a_3 .... a_{13} are the 12 occurrences in ascending order of magnitude, and generalising

MR = T max - T min

where MR = moving range
T = occurrence total

F.3 Range for Branch and Depot Charts

An extension to the control chart technique utilising moving averages (Grant 1980) and moving ranges overcomes the difficulty of comparing ranges of dissimilar vehicle types.

The problem is illustrated below:

Assume we have two vehicle types, with widely varying fuel usages

TABLE F.3.1					
TYPE	USAGE			RANGE	TOTAL
	1	2	3		
A	80	120	100	40	300
B	1000	900	1100	200	3000
DEPOT TOTAL					3300

Therefore Depot Average = 3300/6 = 550

Depot Range = 1100 - 80 = 1020

Obviously the average is somewhat meaningless as an absolute figure. Compared to a similar figure from previous periods however, the variations in this average become meaningful.

The range figure is even less useful since it measures the range of two sets of figures which are totally unrelated.

Now the philosophy of the moving average is discussed in Appendix F.1. If, together with the moving average, a moving range were to be plotted and then in place of the standard control charts for mean and range, control charts for moving average and moving range were plotted, the variations in the range would be those around the total fuel used per month in a branch or depot and not those variations around a number of vehicle types.

This is achieved as follows:

Assume an extension to Table F.3.1 as shown in Table F.3.2

TABLE F.3.2

	MONTH	1	2	3		TOTAL
	1	80	100	90	)	
TYPE [illegible]	2	120	90	100	)	870
		100	100	90	)	
		1000	1050	1100	)	
TYPE [illegible]	2	900	950	900	)	9150
	3	1100	1100	1050	)	
TOTAL		3300	3390	3330		10020

Moving Average = 10020/3 = 3340

Moving Range - 3390 - 3300 = 90

These figures are more meaningful and show the variations to better advantage

The great disadvantage with this method is that it introduces a further concept into the system - one of control charts of moving averages/ranges. Furthermore, the individual points on such charts are not independent of one another and thus in a 3 month moving average chart, 3 points outside the control limits are really equivalent to only one. This obviously reduces the sensitivity of the charts - something one wishes to avoid at this level. After lengthy discussions with the future users of the system, it was decided that since, as far as the control charts were concerned, the most important factors are the charts for the individual vehicles, followed by the mean charts for the branch/depot totals, this would be an unneccessary complication and should not be included in the system.

A simplified, though somewhat unorthodox, [illegible] range chart has therefore been constructed. Since it is not possible (because of space considerations) to keep the figures for each vehicle for each month, the actual figures from which the range was calculated are not available. The range $\bar{R}$ is therefore constructed from a moving range calculation of 2 months (See Grant Pg 297, 1980). This moving range is now utilised in the control charts for mean $\bar{X}$, and range, R, in the normal manner. No reference is made on the range chart to the method used for the calculation of $\bar{R}$ thus removing the extraneous complication in the minds of workshop personnel. Furthermore the use of only 2 months in the moving range calculations limits the reduction in sensitivity of the charts.

Nevertheless, operators should be aware that decisions should not be based entirely on depot or branch control charts but that all the techniques available should be used in conjunction.

APPENDIX G

G.1 Change in usage is slope of the Cusum Chart

The slope of a straight line may be given by:

$$b = \frac{y_2 - y_1}{x_2 - x_1}$$

and assuming that we plot cumulative differences as months, we have for month n:

$$b = \frac{(\text{mean} - \text{act } n) - (\text{mean} - \text{act } n\text{-}_1)}{\text{month } n - \text{month } n\text{-}_1}$$

$$= \frac{\text{act } n\text{-}_1 - \text{act } n}{\text{month } n - \text{month } n\text{-}_1}$$

$$= \text{act } n\text{-}_1 - \text{act } n$$

$$= \text{differences in usage from month } n\text{-}_1 \text{ to month } n$$

Assume a set of occurrences as in Table G.1.

TABLE G.1

Month		1	2	3	4	5
Vehicle	1	80	90	100	90	100
	2	100	90	90	110	90
	3	120	110	100	110	110
Total		300	290	290	310	300
Mean (Historical)		300	300	300	300	300
Difference		0	-10	-10	+10	0
Cumulative Difference		0	-10	-20	-10	-10

Now if the differences are accumulated monthly and plotted on a vertical scale, the result is shown in figure G.1.

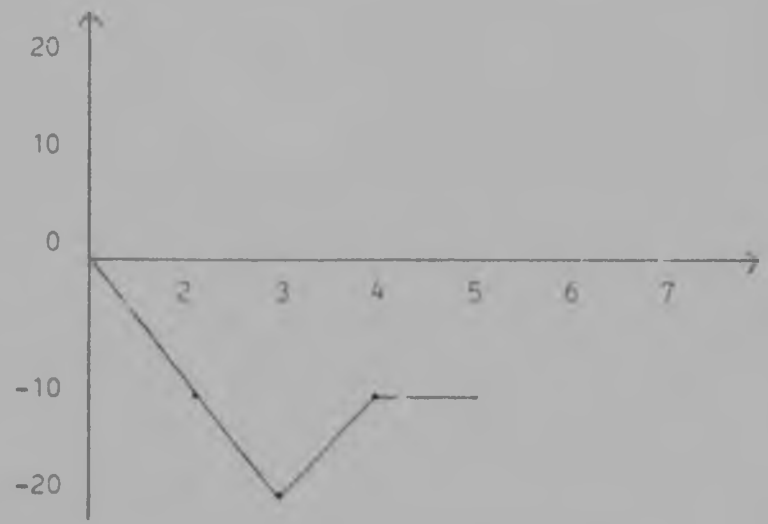


FIG G.1

Note

- The usage in month 2 was the same as that in month 3, as evidenced by the slope.
- The usage in month 4 was different from the mean by the same amount as in month 3 , but with the opposite sign, as can be seen from the slopes of these two sections.
- The usage in month 5 was the same as the mean, as shown by the horizontal slope.

This technique, once understood, may be used to great advantage to show small variations in the mean.

APPENDIX H

H.1 System Breakdown

The following detailed discussion relates to the flow charts shown in figs. H.1 to H.13.

H.1.1 Input

All inputs to the system take the form of data entered through the keyboard. In the case of fuel usage data, this process is somewhat time con: ning but in view of the lack of alternative in the immediate short term, is unavoidable. In the longer term this might be replaced by a procedure, possibly on another machine, whereby the data would be copied from the existing fleetmaster tape onto diskettes, and then utilised by this system.

H.1.1.1 Data Load (Fig H.1)

Data for a specific vehicle type is loaded into the system vehicle by vehicle. The system validates the input and stores the total fuel used by that vehicle type; number of vehicles in the group; and range of fuel usage (difference between highest and lowest). A facility exists to display any data previously loaded and thereafter to make corrections if necessary.

New vehicle types or new depot-branch combinations may be created and the program verifies for prior existence. The program then returns to the main menu.

H.1.1.2 Control Parameter Input (Fig H.2)

The system utilises a control date to monitor file status and print report headings, a moving average spread parameter to define the base period for any moving average graph produced, and a control chart limit parameter in order to define the width of the action limits in the control chart functions.

The moving average may vary between 2 and 24 months, the control chart limit between one and three standard deviations, the control month between one and twelve, and the year between 1981 and 2000.

These may all be changed from the keyboard, any change being validated and echoed for verification before being written to the file. The program returns to the main menu.

H.1.1.3 Default Parameter Input (Fig H.3)

In order to allow the user flexibility of requests as well as to cater for standard non changing requests a system of defaults has been established. These may be changed at will and are used to:

a) Predetermine which of a series of vehicle types within a depot will be printed and which will be excluded when a default type run is requested.

b) Predetermine which of a series of vehicle types will be excluded when accumulating a depot total of fuel usage.

c) Predetermine which of a series of vehicle types will be excluded when accumulating a branch total of fuel usage.

d) Determine whether a particular vehicle type uses petrol or diesel.

These parameters are echoed for verification before being written to the data file. This function also exists in the data load program for the creation of new files.

H.1.1.4 Branch/Depot Load (Fig H.4)

The system uses a series of two digit depot and branch codes as key to the data being loaded.

Each branch and/or depot code loaded onto the system is associated with a description. This description is printed on all output graphs for easy identification.

These descriptions are not validated due to their nature but are echoed for verification before being written to the description master.

H.1.1.5 Coefficient of Range (Fig H.5)

Although the calculation of standard deviation is relatively simple utilising computers (see Appendix A.1) a much simpler method of determining the physical width of statistical control limits makes use of statistical coefficient tables. Such a table is loaded into the system (see Appendix A.5). The coefficients are dependent upon number of vehicles within a vehicle type, i.e. subgroup size and may be altered if necessary. It is not envisaged however, that any change should be made to this file and it is included for completeness. The program returns to the main menu.

H.1.1.6 Load Mean/Range (Fig H.6)

The historical mean and range of each vehicle type is determined by finding the average value of each statistic over a number of months. All months for which accurate data is available should be included. Same measure of caution must be used when doing this since occurrences which were inherently "out of control" could affect the average readings substantially. These should be excluded if they readily can be detected by visual inspection.

The means and ranges for depots and branches are loaded for comparative purposes only and no effort should be made to interpret the absolute values of these figures. This is because the samples making up each occurrence on the graphs (i.e. vehicle types), are non-uniform. These figures may be obtained from the depot and branch data utilising a similar method to that used in calculating individual occurrences (see Appendix F.3).

These figures are loaded into the system using this program and are echoed for verification. The program then returns to the main menu.

H.1.2 Processing

A number of different techniques are utilised to provide useful information on fuel usages. Each is intended to highlight different variances, are complementary, and should be used together if any great degree of certainty is to be achieved in making decisions based on the information.

Each of the following programs require the operator to make a selection on whether the information required is for a single vehicle type, a series of vehicle types within a depot, total of a number of vehicle types within a depot, total of all default types within a depot, or total of all default types within a branch. They also, in certain instances, require the operator to make a decision on whether he is interested in petrol or diesel types.

H.1.2.1 Simple Data Plots (Fig H.7)

This program accumulates per month the data for the chosen combinations and sets up a working storage table containing the past three years data for later plotting. The program then hands control to the plot program, which after plotting the graph, returns to the main menu.

H.1.2.2 Moving Averages (Fig H.8)

The total fuel used is accumulated per month for the combinations selected for the past three years. Working from month one the totals of the first n months are added, where n is moving average spread as defined in the parameter table. This total is then divided by n giving an average for those months. Strictly speaking this average should be applied to the middle month within the spread. However, it is much easier and has become common practice to allocate it to the n TH month. The (n + 1)TH average is calculated by adding to the total of the first n months, the (n + 1) month's total and subtracting the total of month 1, after which this total is divided by n.

n may vary between 2 and 24 months. The size of n has an effect on the number of months data which can be plotted, since the first (n - 1) months are not plotted due to the fact that no average can be established for them.

Once the working storage area has been set up, the program calls the moving average plot program. After plotting the program returns to the main menu.

This technique should be used to smooth wide variations in data to show moving trends in total consumption which would otherwise be hidden within data variations, in order to make comparisons with previous periods. It does not provide detailed information about the data of any one month, but is a simple technique, is easily understood in graphical form and can be used to great advantage for showing total fuel usages.

H.1.2.3 Control Charts (Fig H.9)

The average usage for each vehicle type is determined by dividing the total usage by the number of vehicles in that type for a particular branch-depot combination. The range is taken directly from the history file. The historical mean and the range are extracted from the mean/range master and stored.

H.1.2.3.1 Mean Charts

The data are accumulated per month for the chosen combinations and control limits are then calculated utilising the statistical table and the control limit spread loaded on the parameter file as described in Section 2.4.1. These action line points are stored together with the historical mean and accumulated data for later plotting.

H.1.2.3.2 Range Plots

In the case of single vehicle type requests, the historical range is extracted from the mean/range master and after calculating of control limits, as described in Section 2.4.1, the range is stored together with the monthly data and action line points for later plotting.

In the case of totals per branch and/or totals per depot, a moving range is calculated using the latest two months (see Appendix F.3) and this moving range is then utilised to construct the mean and range charts in the normal manner.

Obviously the above method is not always sufficiently accurate. The range charts for totals should therefore be viewed as a guideline only and extreme care should be taken when making decisions based only on these range charts.

The program now hands over control to the control chart plot program. After plotting the program returns to the main menu. Since the historical mean and range must not change each month, they cannot be calculated from the rolling historical data, and must be manually changed wherever necessary.

Due to space considerations it is not possible to hold the mean and range of every combination of vehicle types, and it is therefore only possible to print control charts for a vehicle type, a series of vehicle types, for the total of all default types within a depot, and for the total of all default types within a branch. Ad hoc combinations are not allowed.

Furthermore it should be noted that accumulated mean and range totals for both depots and branches are not relevant in terms of absolute value, because of the fact that vehicle types making up the totals are totally unalike and will therefore have different means and ranges. The important characteristic on these graphs is the difference between the historical values and the current months calculated values, since this will point out any otherwise unnoticed variations.

Considerable work has been done on the development of various statistical tests for control charts. Appendix A.4 shows how the theory of extreme runs may be applied to detect shifts in a universe parameter. In fact these sequences will occur by chance more often than will a point outside of the 3σ control limits.

Economic factors determine that control charts are best set up so that Type I errors are rare, i.e. an error due to the belief that the universe has changed, when in fact it has not. Control of Type II errors i.e. a mistaken belief that the universe has not changed when in fact it has, may be reduced by increasing subgroup size. Other factors should however be taken into consideration when making this determination (see 2.4.1 and Appendix A.4).

With regard to the testing of a given sample, many statistical tests of hypotheses that samples come from the same or different universes, are available. These are more complex tests and are beyond the scope of elementary control chart principles.

This technique should be used when dealing with specific data points which may vary widely around the mean, where the intention is to investigate only those which do not fall within certain specified limits, as described above. Its advantage lies in the fact that the variations attributable to chance are largely eliminated and only those attributable to other causes need be investigated. It is however, a more complex technique, produces more intricate graphs, and may be less intelligible to the uninitiated. For this reason some care should be taken when introducing the concept to operations personnel.

H.1.2.4 Cusum Charts (Fig H.10)

As with the other techniques, the data is accumulated per month for the combinations selected, into working storage. The figure for each month is then subtracted from the historical mean, extracted off the mean/range master, and the residual added to the previous months cumulative sum thus producing the monthly figure. These cumulative sums for each month are st[illegible]d together with the historical mean for later plotting. Control is handed to the plot program which, after plotting returns to the main menu.

In this technique the change in usage is given by the slope of the line, as can be seen from the example in Appendix G. It will be noticed that a horizontal line indicates a usage exactly equal to the mean, since the figure added to the previous months' cumulative sum is zero and consequently no vertical change occurs. It is thus easy to see when the usage is close to the mean. These charts are extremely useful when attempting to determine very small changes in the process mean, something which cannot be determined from the control chart, since it is easy to detect the change in slope in the "runaway" graph. However the charts do not show absolute values and although a key of slopes is provided such figures are not easily determined from the curve by a layman. Nevertheless in conjunction with the control chart this technique provides an excellent measure of statistical control.

A number of statistical tests are available for use with the cusum technique. These are summarised in Section 2.4.2. However as they stand, these tests are somewhat complex to implement, depending as they do on the extraction of data from the average run length (ARL) curves relevant to the data and test chosen. Wetherill (1977) describes an approximation to these tests, which although strictly speaking requires the use of information from ARL curves, may be arrived at using approximations and is much more practical.

This approach, the Johnson approximate approach, utilises the V-mask technique. The least change in the mean which it is desired to detect with reasonable certainty must be determined. Now the number of standard deviations to be used is given by $\delta\sigma$ and the greatest tolerable probability of false indication by $2\alpha_0$. He now derives the following attributes of the V-Mask.

$\sigma = \tan^{-1} \delta/_4$

$d = -2\log_e(\alpha_0)/\delta^2$

The mask may now be easily drawn and any points falling outside of this mask would indicate a change in mean.

However, even this simplification, in the everyday use of these charts, is not easily understandable by the uninitiated, and it has become common practice to simply plot an average slope line on the chart, critically evaluating it to determine whether any change is sufficient to warrant investigation.

As with control charts, cusum charts may only be printed for a vehicle type, a series of vehicle types, for the total of all default types within a depot, and for the total of all default types within a branch. Also, as with control charts, the absolute values of the means of total usage per depot and per branch are not important. The differences between the means and the monthly data i.e. the slope of the curve is the important parameter.

H.1.2.5 File Roll (Fig H.11)

Because of the limited space availabe on the diskettes utilised in this system, the decision was taken to hold no more than 36 running months of data. Some function is required therefore to enable the roll-off of the 36th month, each month. This program reads the data for the requested combination, drops off the 36th month back, and overwrites each month's data with the data of the following month, effectively rolling each month back by one month. The new month's data may now be loaded into the current month.

The facility is provided to roll each depot individually or to roll the entire branch together, in the event that it is desired to load data for two different months on two different depots. This procedure is however not recommended since it places on the user the responsibility of ensuring that all depots have been rolled before printing any graphs. This arises because the date printed on the graphs is the control date which may only be changed after all files have been rolled.

H.1.3 Output

Output from the system consist mainly of graphical representation of data. The following types are produced:

H.1.3.1 Simple Data Graphs (Fig H.7)

These graphs are produced individually, in sequence, or in totals as determined in H.1.2.1. Descriptions for branch and depot codes are read from the branch/depot master and the date is taken from the control file.

The limits and scales on the axes are determined from the data itself thus obviating the necessity of operator input, while a table is printed of all vehicle types accumulated into the graph shown.

H.1.3.2 Moving Average Charts (Fig H.8)

These graphs are produced individually, in sequence, or in total as determined in H.1.2.2. Descriptions, date, vehicle types and scales are as in H.1.3.1. Furthermore the moving average spread is shown in the heading. In view of the fact however, that the first n - 1 months (i.e. the spread) have no average associated with them, the graphs will show a period somewhat less than three years, depending on the spread.

H.1.3.3 Control Chart Plots (Fig H.9)

Since it was necessary to fit two plots onto one page, one below the other, the displayd period of data has been shortened to two years. This, it is believed, is still sufficient to show an accurate picture of the existing situation.

Descriptions, vehicle types, date and the number of standard deviations used to draw the control limits are shown. Scales are automatically chosen by the system to fit within the scaled boundaries and to calculate axis limits.

Charts may be plotted individually, for a series, or for a total as determined in H.1.2.3.

H.1.3.4 Cusum Plots (Fig H.10)

These charts may be plotted individually, in sequence or in totals, as with the others. Descriptions, scales, date and vehicle types are shown.

Theoretically the most practical scale is where one unit on the x- axis equals two standard deviations on the y- axis. However the dimensions of the available plotter makes this difficult in some instances and it has been decided to allow the scale to vary with the nature of data, identification of the slope being provided by a key showing the registration of all major angles on the chart.

H.1.3.5 Print of Data (Fig H.12)

In order to provide a hard copy of the data loaded in the system at any time for any given vehicle type, depot total or branch total, a listing may be produced detailing for each month, consumption figures of petrol and diesel together with year-to-date figures of each for comparative purposes.

These, although not intended to be used for statistical analysis, should provide useful supporting information if required.

H.1.3.6 Print of Depots/Branches (Fig H.13)

A table of all existing depot and branch codes together with their descriptions may be printed. Since these codes are used extensively throughout the system some easy reference is necessary should it be required.

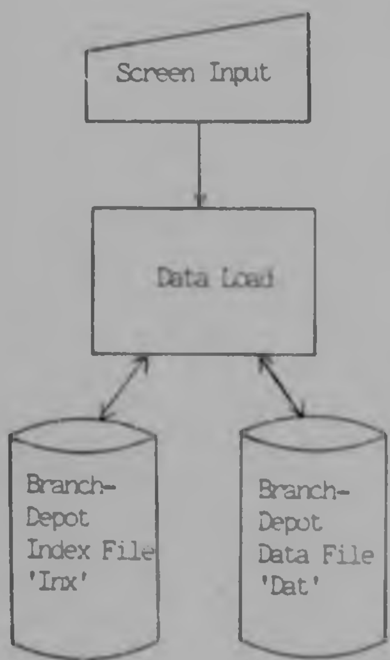


FIG H.1

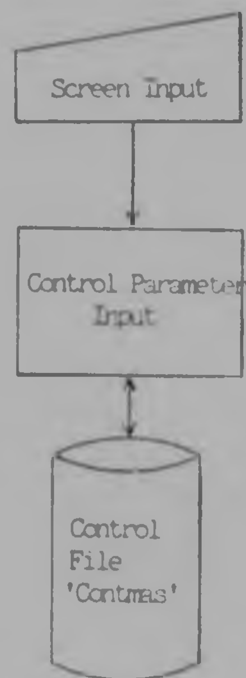


FIG H.2

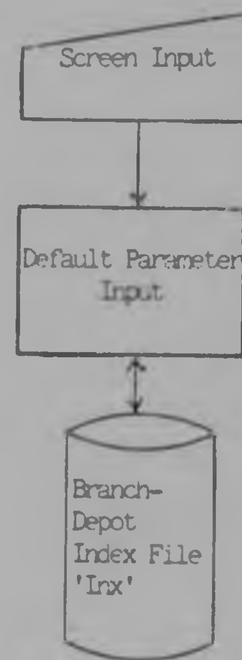


FIG H.3

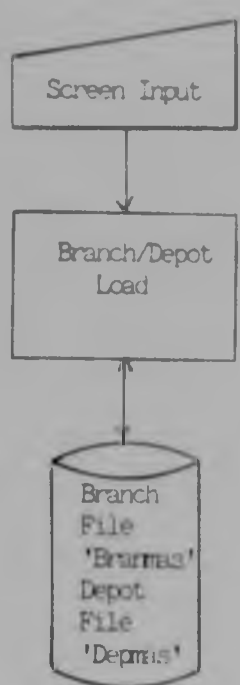


FIG H.4

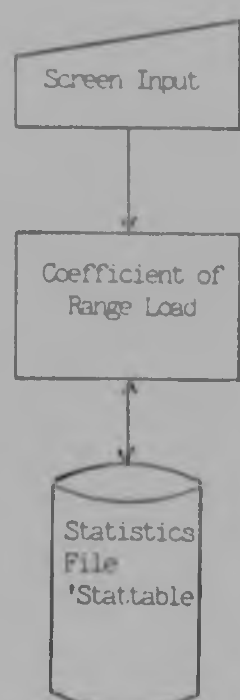


FIG H.5

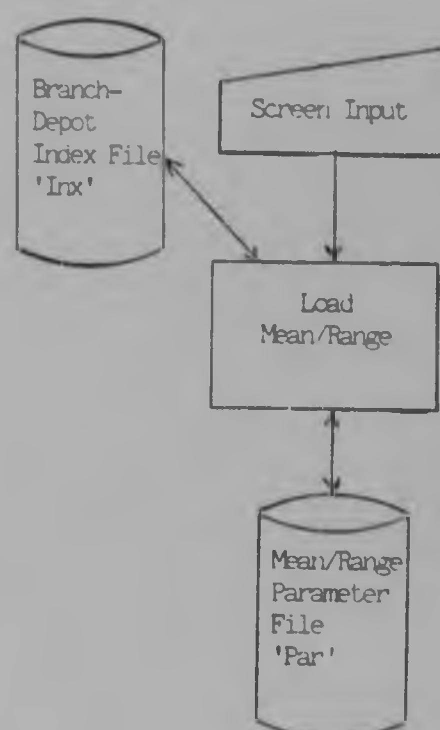


FIG H.6

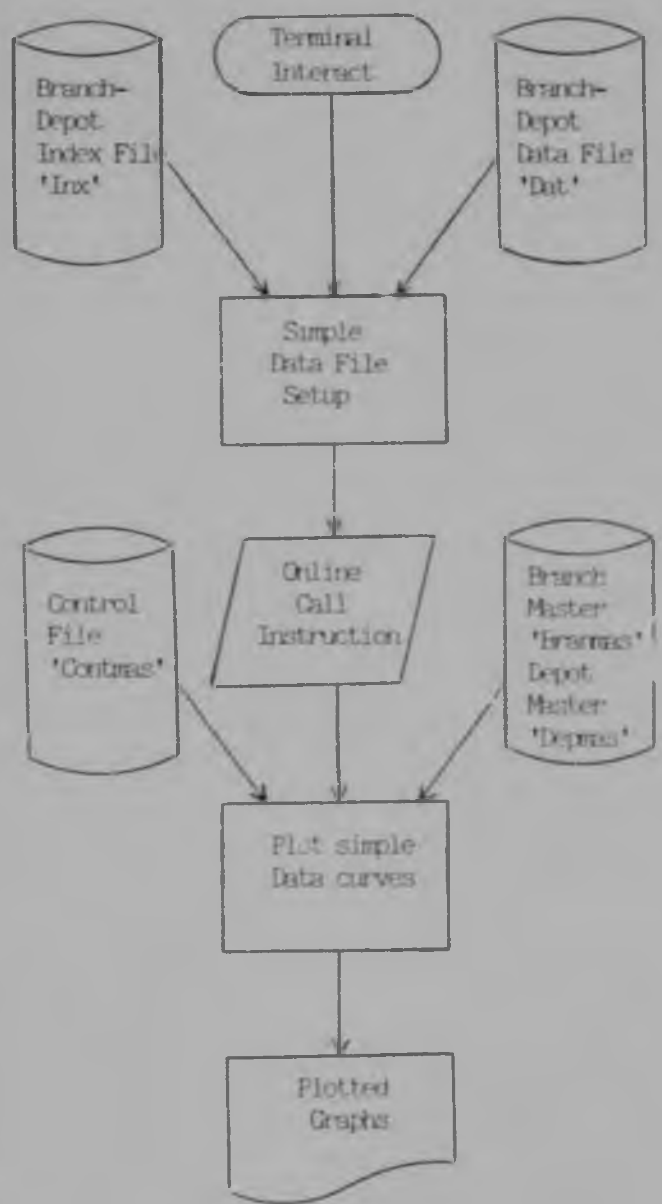

Branch-
Depot
Index File
'Inx'
Terminal
Interact
Branch-
Depot
Data File
'Dat'
Simple
Data File
Setup
Control
File
'Contmas'
Online
Call
Instruction
Branch
Master
'Branmas'
Depot
Master
'Depmas'
Plot simple
Data curves
Plotted
Graphs

FIG H.7

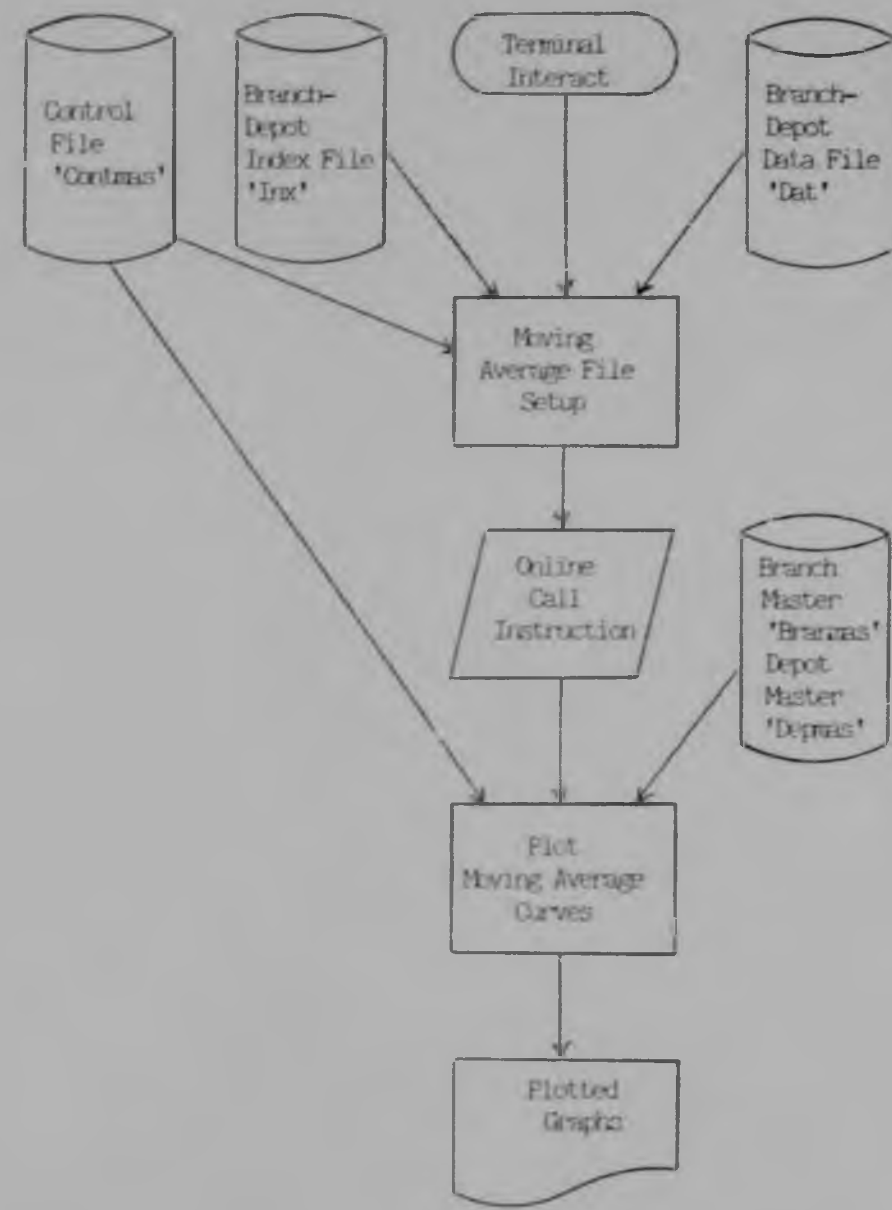

Control
File
'Contmas'
Branch-
Depot
Index File
'Inx'
Terminal
Interact
Branch-
Depot
Data File
'Dat'
Moving
Average File
Setup
Online
Call
Instruction
Branch
Master
'Branmas'
Depot
Master
'Depmas'
Plot
Moving Average
Curves
Plotted
Graphs

FIG H.8

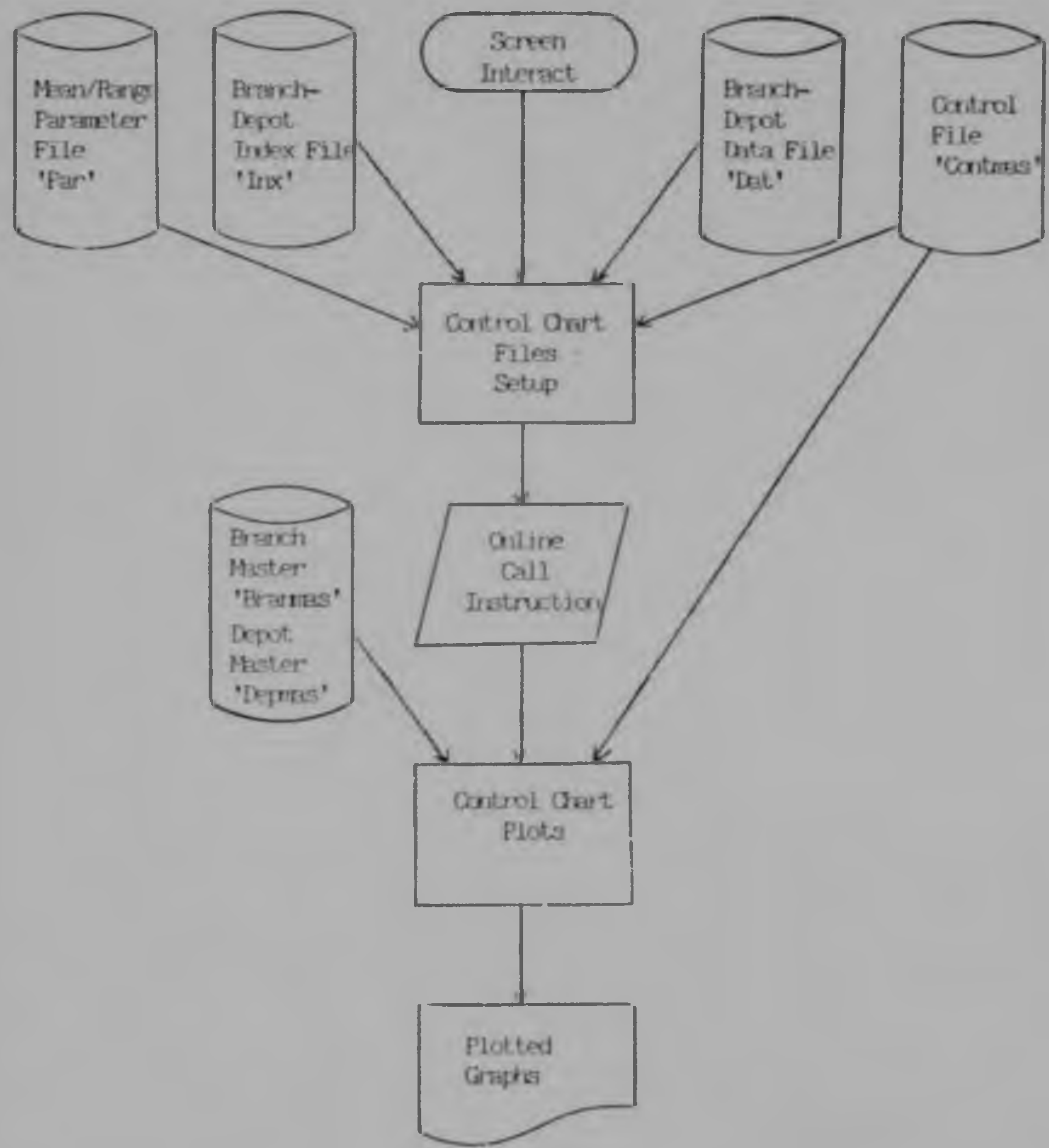


FIG H.9

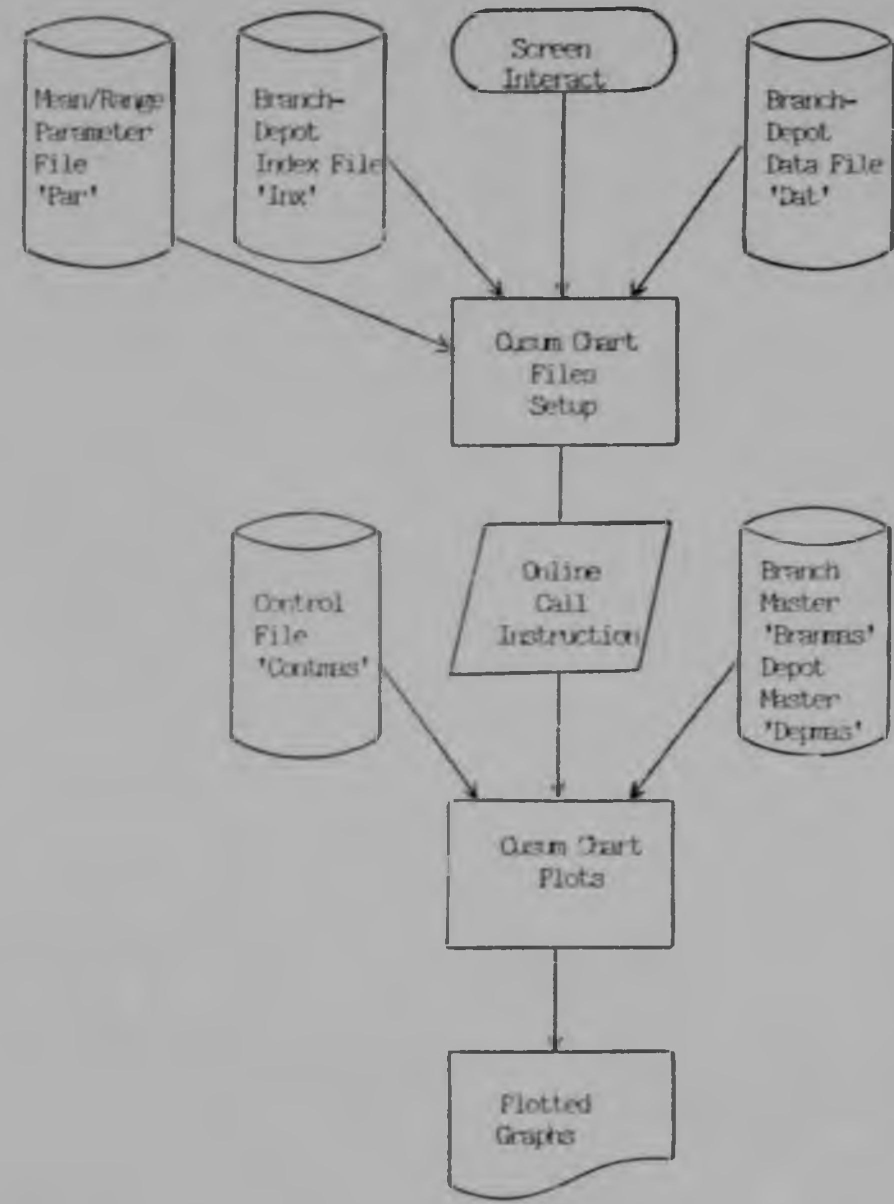


FIG H.10

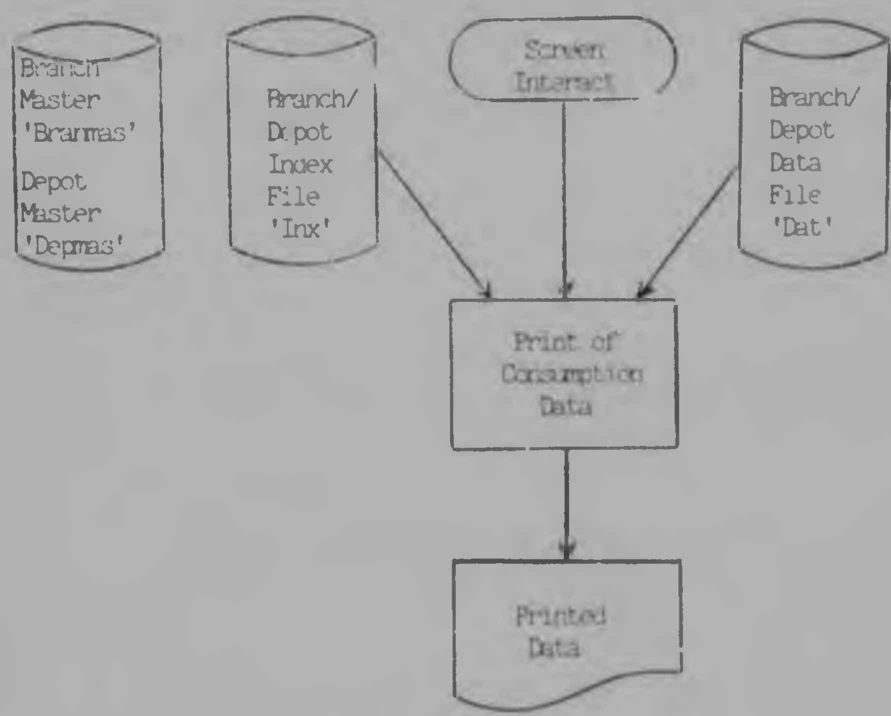


FIG H.12

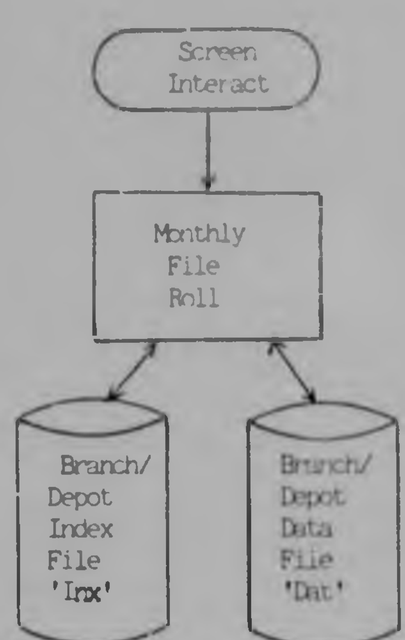


FIG H.11

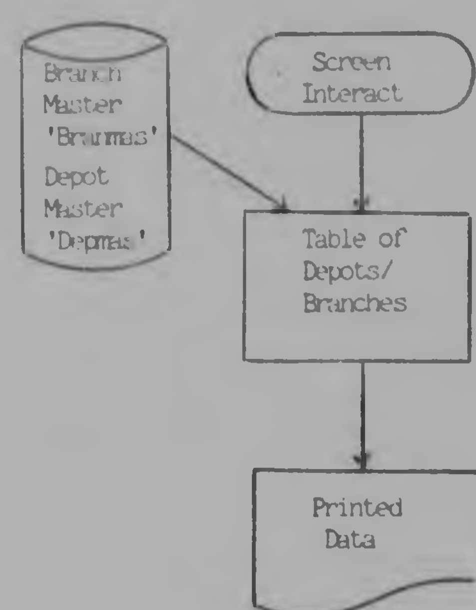


FIG H.13

REFERENCES

BARTLETT [illegible] (1980) — Fuel Efficiency [illegible] 198[illegible] Truck Meas[illegible]
Public Works November 1980 P61-6[illegible]

BASCUNANA [illegible] and STAHMAN R.[illegible] (1978) — [illegible]uel Characteristics Affect Economy
Automotive Engineering Vol 86 N° [illegible]
[illegible]54-55

[illegible] — [illegible]ehicle Mission Simulat[illegible]
Systems June 1983 P15-1[illegible]

BOX G.E.P. and JENKINS G.[illegible] (19[illegible]) — Time Series Analysis Forecasting [illegible]
Control
Holden Day, San Fransisco, U.S.A.

BRAVERMAN [illegible] (19[illegible]) — Fundamentals of Statistical Quality
Control
Reston Publishing. Virginia, U.S.A.

CAPOR[illegible] — Statistical and Data Analysis
Programs for Minicomputers
General Electric Company, Schenectady,
N.[illegible]

CHATFIELD C. (1978) — Statistics for Technology 2nd Ed.
Chapman and Hall. New York

CHIU W.K. and WETHERILL G.B. (1974) — A Simplified Scheme for the Econom[illegible]
Design of $\bar{X}$ Charts
Journal of Quality Technology
Vol 6 N° 2 P63-69

[illegible] — [illegible]icrocomputer Based Vehicl[illegible]
[illegible]nagement Information System
University of the Witwatersrand,
Faculty of Engineering 120 pages

Statistics 4th Ed
Richard D Irwin, Homewood Illinois,
U.S.A

EMSHOFF ... and SISSON R.L. 1970 — Design and Use of Computer Simulation Models
MacMillan Publishing Co., New York

FLACHSBART — Evaluating Motor Fuel Conservation Policies at the Community Level
Traf... arterly Vol 33 N° ...

GRANT E. ... and LEAVENWORTH R. ... — Statistical Quality Control ...
McGraw - Hill. Tokyo

GREENWOOD J.A. and HARTLEY H.D. (1962) — Guide to Tables in Mathematical Statistics
Princeton University Press.
New Jersey

... — How Does Ambient Temperature Affect ... conomy and Emissions
Automotive Engineering
Vol 87 N 1 P 78-80

HEIKES R.G. ... MONTGOMERY ... — ...ductivity is Enhanced ... Statistical Quality Control
Industrial Engineer Vol 12 P ...

HOTARD D.G. and JORDAN J.D. (1981) — Regression Analysis is Applied Improve Product Quality
Industrial Engineer vol 19 P 68-75

HUITSON A. and KEEN J. (1965) — Essentials of Quality Control
Heinemann, London

JANSSEN L.J. (1978) — Aerodynamic Improvement - Better Fuel Economy
Society of Automotive Engineers Preprints N° 780265

KREFT I.J. (1980) — Control Charts Help Set Firms' Energy Management Goals
Industrial Engineer Vol 19 P 56-58

LA POINTE C. (1973) — Factors Affecting Vehicle Fuel Economy
Proceedings of FCIN and Fuels and Lubricants Meeting
September 1973, Paper 730791
P 100-110

MONTGOMERY D.G. (1980) — The Economic Design of Control Charts : A Review and Literature Survey
Journal of Quality Technology
Vol 12 N° 2 P 75-87

MURDOCH J. (1979) — Control Charts
MacMillan Press, London

RATCLIFFE B. (1980) — Fuel Conservation : Much Talk but Not a Lot of Action
Transport July/August 1980 P 55-57

RIGGS J.L. — Production Systems : Planning, Analysis and Control 2nd Ed
Wiley Series in Management

SARKADI K. and VINCZE I. (1974) — Mathematical Methods of Statistical Quality Control
Academic Press, New York

SHEWHART W.A. (1931) Economic Control of Quality of Manufactured Product
5th Printing, Van Nostrand, N.Y.

SMILEY C. (1983) Driver Training Program Can Cut Operating Costs
World Wastes July 1983, P 30-32

SMITH G.E.D. (1983) Transport Management Software Update
Systems June 1983 P 22-23

SWEET L.M. (1981) Control systems for Automotive Vehicle Fuel Economy - A Literature Review
Journal of Dynamic Systems, Measurement Control
Vol 103 P 173-180

TAYLOR H.M. (1965) Markovian Sequential Replacement Processes
Annals of Mathematical Statistics
Vol 36 P 1677-1694

WATERS M.H.L. (1980) Research on Energy Conservation for Cars and Goods Vehicles
Transport and Road Research Laboratory Report 591
Crowthorne, 12 Pages

WETHERILL G.B. (1977) Sampling Inspection and Quality Control
2nd Ed Chapman and Hall, London

ANON (1983) The One-Stop Fleet Control System
Systems
June 1983 P 18-20

ANON (1983) Fuel System Proven Efficient for Fleet
World Wastes July 1983, P28-29

INVESTIGATION INTO THE SYSTEM OF FUEL MANAGEMENT IN THE JOHANNESBURG CITY ENGINEERS DEPARTMENT AND THE DESIGN OF A MICROPROCESSOR BASED SYSTEM TO ASSIST IN THE CONTROL THEREOF.

DALE STEVEN DRUCKMAN

- VOLUME II

A dissertation submitted to the Faculty of Engineering, University of the Witwatersrand for the degree of Master of Science in Engineering.

JOHANNESBURG 1984

11. PROGRAM SOURCE LISTINGS

The following pages contain source listings for each program used in the system. These are intended for reference only and are included for completeness.

```
1    REM   ***************   PROGRAM MAINMENU   *******************.
2    REM   **  THIS PROGRAM PROVIDES ALL THE FUNCTIONS AVAILABLE  **
3    REM   **  IN THE SYSTEM.ALL PROGRAMS RETURN TO THIS MENU.     **
4    REM   ********************************************************
10   CLEAR
11   DISP
20   DISP ,"***  M A S T E R   M E N U  ***"
30   DISP
40   DISP
50   DISP
60   DISP "            1. LOAD DATA.                 8. CHANGE DEPOTS."
70   DISP "            2. SIMPLE DATA PLOTS.         9. CONTROL PARAMETERS."
80   DISP "            3. MOVING AVERAGES.          10. STATISTICS TABLE."
90   DISP "            4. CONTROL CHARTS.           11. FILE ROLL."
100  DISP "            5. PRINT DATA.               12. DEFAULT MAINTENANCE."
110  DISP "            6. LOAD MEAN/RANGE.          13. PRNT BRNCH/DEF CODE LIST."

115  DISP "            7. CHANGE BRANCHES.          14. CUSUM CHARTS."
120  DISP "                              15. END."
121  DISP "MAKE YOUR SELECTION"
130  INPUT ANSWER
140  ON ANSWER GOTO PARA1 ,PARA2 ,PARA3 ,PARA4 ,PARA5 ,PARA6 ,PARA7 ,PARA8 ,PARA9
 ,PARA10 ,PARA11 ,PARA12 ,PARA13 ,PARA14 ,PARA15
150 PARA1: CHAIN "DATALOAD"
160 PARA2: CHAIN "SIMPDAT"
170 PARA3: CHAIN "MOVINGAVE"
180 PARA4: CHAIN "CONTCHART"
190 PARA5: CHAIN "DATAPRNT"
200 PARA6: CHAIN "PARAMMNT"
210 PARA7: CHAIN "BRANMAINT"
220 PARA8: CHAIN "DEPOTMAINT"
230 PARA9: CHAIN "CONTMAINT"
240 PARA10: CHAIN "STATTABMNT"
242 PARA11: CHAIN "DATAROLL"
244 PARA12: CHAIN "DEFMAINT"
245 PARA13: CHAIN "LISDEPBRN"
250 PARA14: CHAIN "CUSUM"
255 PARA15: CHAIN "CLOSEOFF"
260 STOP
270 REM  ************  PROGRAM MAINMENU TERMINATE  ****************
```

```
10   REM  **********   PROGRAM  DATALOAD  ******************
20   REM  **  THIS PROGRAM ACCEPTS VEHICLE FUEL TOTALS     **
30   REM  **  AND CALCULATES TOTAL FUEL USED.NO OF         **
40   REM  **  VEHICLES AND RANGE OF TOTALS PER TYPE        **
50   REM  **  WITHIN A SPECIFIC BRANCH AND DEPOT, AND      **
60   REM  **  THEN WRITES THEM TO DISC.                    **
70   REM  **  IT ALSO ALLOWS INSPECTION OF ANY TOTALS      **
80   REM  **  LOADED IN PREVIOUS MONTHS.                   **
90   REM  ***********************************************
100          OPTION BASE 1
110          DIM TOTAL(36,3),FLEETINF(20,3)
120 BRNCH: FOR K=1 TO 36
130              FOR L=1 TO 3
140                  TOTAL(K,L)=0
150              NEXT L
160          NEXT K
161 CLEAR
162         DISP
163         DISP ,"  ***  DATA LOAD FUNCTION  ***"
164         DISP
165         DISP
170          DISP "WHICH BRANCH"
180          INPUT BRANCH
190          DISP "WHICH DEPOT"
200          INPUT DEPOT
210  FLTYP: DISP "WHICH TYPE "
220          DISP "(TYPE CODE OR 0 TO END)"
230          INPUT TYPE
235          IF TYPE<0 OR TYPE>100 THEN DISP "TYPE INVALID" @ GOTO FLTYP
240         IF TYPE=0 THEN GOTO FINI  ELSE NEWFL
250 NEWFL: DISP ,"YOU ARE INTERESTED IN "
260         DISP ,"BRANCH ";BRANCH;" DEPOT ";DEPOT;" TYPE ";TYPE
270         DISP ,"IS THIS CORRECT(Y/N)"
280         INPUT VALID1$
290         IF VALID1$="N" THEN GOTO BRNCH
291         IF VALID1$ <> "N" AND VALID1$ <> "Y" THEN GOTO NEWFL
292         BRANCH$=VAL$ (BRANCH)
293         DEPOT$=VAL$ (DEPOT)
300 FLNEW: DISP "IS THIS A NEW FILE(Y,N)"
310         INPUT FLANS$
319         IF FLANS$ <> "Y" AND FLANS$ <> "N" THEN GOTO FLNEW
320         IF FLANS$="Y" THEN GOTO NEWCR  ELSE GOTO READFL
325 NEWCR: ON ERROR GOTO ERR1
326         DISP ," NEW FILE CREATION - APPROX 3 MINUTES"
330         CREATE BRANCH$&DEPOT$&"INX:D700",100,20
340         CREATE BRANCH$&DEPOT$&"DAT:D700",50,872
350         ASSIGN# 1 TO  BRANCH$&DEPOT$&"INX:D700"
360         ASSIGN# 2 TO  BRANCH$&DEPOT$&"DAT:D700"
370         OFF ERROR
380 ZERISE: FOR P=1 TO 100
390         POSITION=0
395         IF P=100 THEN POSITION=1
400         EXCINC$="YYY"
401         FUEL$="N"
410         PRINT# 1,P ; POSITION,EXCINC$,FUEL$
420         NEXT P
421         FOR P=1 TO 50
422         IND=0
423         PRINT# 2,P ; IND,TOTAL(,)
424         NEXT P
430         GOSUB SETEXCINC
431         POSITION=1
432         PRINT# 1,TYPE ; POSITION,EXCINC$,FUEL$
433         IND=1
```

```
            PRINT# 2,POSITION ; IND,TOTAL(,)
            GOTO LDDATA
 READFL: ON ERROR GOTO
             ASSIGN# 1 TO  BRANCH# DEPOT#
             ASSIGN# 2 TO  BRANCH# DEPOT# DAT:D
             OFF ERROR
             READ# 1,TYPE ; POSITION,EXTLINC$,FUEL$
                     IF POSITION=0 THEN GOTO NEWSRCH
            READ# 2,POSITION ; IND,TOTAL(,)
 PRVDAT: DISP "DO YOU WISH TO SEE PREV DATA(Y/N)
             INPUT RESP1$
            IF RESP1$="N" THEN GOTO CHKROL
            IF RESP1$   "N" AND RESP1$      THEN GOTO PRVDAT
 DATANS: DISP "HOW MANY MONTHS BACK"
            DISP "(LATEST MONTH = 1,PREV MONTH =      ,ETC)
            INPUT MONTH
            IF MONTH=0 THEN GOTO DISPALL  ELSE DISMTH
 DISPALL: CLEAR
              DISP "MONTH","FUEL","NO OF VEH","RANGE"
            FOR I=1 TO 36
            DISP 37-I,TOTAL(I,1),TOTAL(I,2),TOTAL(I,3)
            IF I=12 OR I=24 THEN GOTO NEWPG  ELSE NXTI
 NEWPG: DISP "MORE DATA(Y/N)"
            INPUT MORRESP$
            IF MORRESP$="N" THEN GOTO MORDAT
            CLEAR
            DISP "MONTH","FUEL","NO OF VEH","RANGE
 NXTI: NEXT I
            DISP "OK(Y/N)"
            INPUT OKRESP$
            IF OKRESP$#"Y" THEN GOTO DISPALL
            CLEAR
            GOTO MORDAT
 DISMTH: LET I=37-MONTH
            DISP "TOTAL FUEL USED   ";TOTAL(I,1)
            DISP "NO OF VEHICLES    ";TOTAL(I,2)
            DISP "RANGE             ";TOTAL(I,3)
            DISP
 MORDAT: DISP "DO YOU WISH TO SEE MORE DATA(Y/N)
            INPUT RESP1A$
            IF RESP1A$="Y" THEN GOTO DATANS
            IF RESP1A$    "Y" AND RESP1A$    "N" THEN GOTO MORDAT
 CHKROL: DISP "HAVE YOU ROLLED THE FILE(Y/N)
            INPUT RESP2$
            IF RESP2$="Y" THEN GOTO LDDATA
            IF RESP2$    "Y" AND RESP2$    "N" THEN GOTO CHKROL
            DISP "**  PLEASE NOTE THAT ANY DATA PUNCHED TO THE **"
            DISP "**  LATEST MONTH ON THIS FILE WILL OVERWRITE **"
            DISP "**  EXISTING DATA !!                         **"
            DISP "DO YOU WISH TO CONTINUE(Y/N)"
            INPUT RESP2A$
            IF RESP2A$="Y" THEN GOTO LDDATA  ELSE DISP "PLEASE PERFORM THE ROLL F
UNCTION ON THIS FILE" @ WAIT 5000 @ GOTO FINI
 LDDATA: DISP "WHICH MONTHS DATA DO YOU WISH TO LOAD"
             DISP "(LATEST MONTH = 1,PREV MONTH = 2,ETC)"
             INPUT RESP3A
             MONTH=37-RESP3A
             LET J=1
 DATFLT: DISP "ENTER VEHICLE FLEETNO"
            INPUT FLEETINF(J,1)
            DISP "ENTER FUEL USED"
            INPUT FLEETINF(J,2)
 MORVEH: DISP "ARE THERE MORE VEHICLES IN THE FLEET(Y/N)"
            INPUT RESP4$
            IF RESP4$    "Y" AND RESP4$    "N" THEN GOTO MORVEH
            IF RESP4$="Y" THEN GOTO INCCNT  ELSE GOTO CLCTOT
```

```
4?4         PRINT# [illegible]
440         GOTO LDDATA
450 RE?DF   ON ERROR GOTO ERR2
460          ASSIGN# 1 TO  BRANCH$ [illegible] "IN [illegible]
470          ASSIGN# 2 TO  BR[illegible] OTI  D[illegible]
475          OFF ERROR
480          READ# 1,TYPE ; POSITION,EXCINC$, UE[illegible]
490               IF POSITION[illegible] THEN GOTO NEWSRCH
500         READ# 2,POSITION ; IN[illegible]OTAL(,)
510 PRVDAT: DISP "DO YOU WISH TO SEE PRE[illegible] DATA  [illegible]N)"
520          INPUT RESP1$
530         IF RE[illegible]1$ "N" THEN GOT  CH[illegible]OL
531         IF RESP1$    "N" AND RESP1$    "Y" THEN GOTO PR[illegible]DAT
540 DATANS: DI[illegible]P "HOW MANY MONTH[illegible]
550         DISP "(LATEST MONTH = 1,PREV MONTH = 2,ALL = 0,ETC)"
560         INPUT MONTH
570         IF MONTH=0 THEN GOTO DISPALL  ELSE DISMTH
581 DISPALL: CLEAR
590           DISP "MONTH","FUE[illegible]"NO OF VEH","RANG[illegible]
600         FOR I [illegible]1 TO ?6
610         DISP 3[illegible] I,TOTAL(I,1[illegible],T[illegible]L(I,[illegible]),TOTAL(I,[illegible]
611         IF I=12 OR I[illegible]24 THEN GOTO NE[illegible]F[illegible]  ELSE NXTI
61? NEWFG: DISP "MORE DATA(Y[illegible]
61?         INPUT MORRESP$
614         IF MORRESP[illegible]"N" THEN GOTO M[illegible]
615         CLEAR
616         DISP "MONTH","FUEL", NO [illegible]H","RANGE
620 NXTI: NEXT I
621         DISP "OK(Y/N)"
622         INPUT OKRESP$
623         IF OKRESP$#"Y" THEN GOTO DISPALL
624         CLEAR
640         GOTO MORDAT
650 DISMTH: LET I=[illegible]-MONTH
660         DISP "TOTAL FUEL USED =";TOTAL(I,1)
670         DISP "NO OF VEHICLES  =";TOTAL(I,2)
680         DISP "RANGE               =";T[illegible]AL(I,[illegible]
690         DISP
700 MORDAT: DI[illegible]P "DO YOU WI[illegible]H T[illegible] MORE DA[illegible](Y N)"
710         INPUT RESP1A$
720         IF RESP1A$="Y" THEN GOTO DATANS
721         IF RESP1A$    "Y" AND [illegible]ESP1A$    "N" THEN GOTO MORDAT
730 CHROL: DISP [illegible]HAVE [illegible]OU ROLL[illegible]D TH[illegible] FIL[illegible]Y/N)"
740         INPUT RESP2$
750         IF RESP2$="Y" THEN GOTO LDDATA
7?1         IF RE[illegible]$    "Y" AND [illegible]$    "N" THEN GOTO CHROL
760         DISP "** PLEASE NOTE THAT ANY DATA PUNCHED TO THE **"
770         DISP "** LATEST MONTH ON THI[illegible] FILE WILL OVERWRITE **"
77?         DISP "** EXISTING DATA !!!                          **"
780         DISP "DO YOU WISH TO CONTINUE(Y/N)"
790         INPUT RESP2A$
800         IF RE[illegible]A$="Y" THEN GOTO LDDATA  ELSE DISP "PLEASE PERFORM [illegible]
UNCTION ON THIS FILE" [illegible] WAIT 20[illegible] @ GOTO FINI
810 LDDATA: DISP "WHICH MONTHS DATA DO YOU WISH TO LOAD";
820          DISP "(LATE[illegible]T MONTH = 1,[illegible]EV MONTH = 2,ETC)"
830          INPUT RESP3A
840          MONTH[illegible]-RESP3A
850          LET J=1
860 DSPFLT: D[illegible]SP "ENTER VEHICLE FLEETNO"
870         INPUT FLEETINF(J,1)
880         DISP "ENTER FUEL USED"
890          INPUT FLEETINF(J,2)
900 MORVEH: DISP "ARE THERE MORE VEHICLES IN THE FLEET(Y/N)"
910         INPUT R[illegible]$
919         IF RESP[illegible]$    "[illegible] AND [illegible]$    "N" THEN GOTO MORVEH
920         IF RESP[illegible]$="Y" THEN GOTO INCCNT  ELSE GOTO CLCTOT
```

```
920 [illegible]
940         GOTO [illegible]
950 CLC[illegible]: [illegible]
960         [illegible]
970         [illegible]
980         [illegible]
990         [illegible]
1000        FOR I=1 TO [illegible]
1010            [illegible]
1020        NEXT I
1030        [illegible]
1040        FOR J=1 TO [illegible]
1050            [illegible]
1060            [illegible]
1070        NEXT J
1080        [illegible]
1090 WRTE[illegible]: PRINT# [illegible]
1100 [illegible]: DISP "DO YOU WISH TO [illegible]
1110        INPUT [illegible]
1115        IF [illegible] THEN GOTO [illegible]
1120        IF [illegible] THEN [illegible] ELSE GOTO FINI
1130 FINI: [illegible]
1140        [illegible]
1150        DISP [illegible]
1152        [illegible]
1160        END
1170 [illegible]: CLEAR
1180        DISP [illegible] LOAD A [illegible] VEHICLE FILE"
1190        DISP [illegible]
1200        DISP [illegible] CORRECT [illegible]
1210        INPUT [illegible]
1220        IF [illegible] THEN GOTO [illegible]
1230 [illegible]: FOR [illegible]
1240        [illegible]
1250        IF [illegible] THEN GOTO CHECK
1260        NEXT [illegible]
1270        DISP [illegible]
1280        GOTO FINI
1285 CHECK: [illegible]
1290        [illegible]
1300        [illegible]
1310        [illegible]
1320        [illegible]
1330        GOTO [illegible]
1340 [illegible]: CLEAR
1341 [illegible]
1342 DISP
1350 [illegible]: DISP [illegible]
1360            [illegible]
1370            INPUT [illegible]
1371            IF [illegible] THEN GOTO [illegible]
1380 [illegible]: DISP [illegible]
1390            [illegible]
1400            INPUT [illegible]
1401            IF [illegible] THEN GOTO [illegible]
1410 [illegible]: DISP [illegible]
1420            [illegible]
1430            INPUT [illegible]
1431            IF [illegible] THEN GOTO [illegible]
1440            [illegible]
```

```
1480 ERR.   [illegible]
1490        WAIT 3000 @ GOTO [illegible]
1500        DISP [illegible] FILE [illegible]
[illegible] WAIT 3000 @ GOTO ERNCL
```

```
10 REM   ********************  PROGRAM  CONTMAINT   ********.*** ********
20 REM   ***   THIS PROGRAM IS USED TO MAKE CHANGES TO THE DIFFERENT      ***
30 REM   ***   CONTROL PARAMETERS USED IN THE SYSTEM.                     ***
40 REM   ***   THE CONTROL LIMIT PARAMETER CHANGES THE WIDTH OF THE       ***
50 REM   ***   LIMITS ON THE CONTROL CHART(NORMALLY SET TO 3)             ***
60 REM   ***   THE DATE PARAMETER IS TO BE USED FOR PRINTING AND          ***
70 REM   ***   PLOTTING PURPOSES.                                         ***
80 REM   ***   THE MOVING AVERAGE PERIOD DETERMINES THE NO OF MONTHS      ***
90 REM   ***   OVER WHICH THE AVERAGE IS TAKEN.                           ***
100 REM  ******************************************************************
110 CLEAR
130 ASSIGN# 9 TO  "CONTMAS:D700"
138 DISP
139 INITL: DISP
140 DISP ,"*** CONTROL FILE MAINTENANCE ***
150 DISP
160 DISP
170 DISP ,"1. CHANGE CONTROL LIMITS."
180 DISP ,"2. CHANGE CONTROL DATE."
190 DISP ,"3. CHANGE MOVING AVERAGE PERIOD."
200 DISP
205 READ# 9,1 : CONTLIM,MONTH,YEAR,SPREAD
210 DISP "PLEASE MAKE YOUR SELECTION"
220 INPUT1: INPUT RESP1
230          IF RESP1 1 OR RESP1 3 THEN GOTO INPUT1
240          ON RES 1 GOSUB CNTLIM ,DATCHG ,MOVAVE
250 FINI: ASSIGN# 9 TO  *
260        DISP ,"PROCESSING COMPLETED"
265        CHAIN "MAINMENU"
270        STOP
280 CNTLIM: DISP "THE CURRENT CONTROL LIMITS ARE SET TO:"
281          DISP ,CONTLIM;"SIGMA"
290          DISP "PLEASE TYPE IN THE NEW CONTROL LIMIT"
300          INPUT WCONTLIM
301          IF WCONTLIM 1 OR WCONTLIM 4 THEN GOTO ERR1
302 NEWLIM: DISP "THE NEW LIMIT IS :"
303          DISP WCONTLIM;"SIGMA"
304          DISP "IS THIS CORRECT(Y/N)"
305          INPUT RESP9$
306          IF RESP9$    "Y" AND RESP9$    "N" THEN GOTO NEWLIM
307          IF RESP9$= N" THEN GOTO CNTLIM
308          CONTLIM=WCONTLIM
310          PRINT# 9,1 : CONTLIM,MONTH,YEAR,SPREAD
320          RETURN
330 DATCHG: DISP "THE CURRENT CONTROL DATE IS:"
331          DISP ,MONTH;YEAR
340          DISP "PLEASE TYPE IN THE NEW DATE IN FORMAT MONTH,YEAR"
350          INPUT WMONTH,WYEAR
351          IF WYEAR 1980 OR WYEAR . 00 THEN GOTO ERR2
352          IF WMONTH 1 OR WMONTH 12 THEN GOTO ERR3
353 NEWDAT: DISP "THE NEW CONTROL DATE IS:"
354          DISP ,WMONTH;WYEAR
355          DISP "IS THIS CORRECT(Y/N)"
356          INPUT RESP8$
357          IF RESP8$    "Y" AND RESP8$    "N" THEN GOTO NEWDA.
358          IF RESP8$="N" THEN GOTO DATCHG
359          MONTH=WMONTH
360          YEAR=WYEAR
361          PRINT# 9,1   CONTLIM,MONTH,YEAR,SPREAD
370          RETURN
380 MOVAVE: DISP "THE CURRENT MOVING AVERAGE SPREAD IS:
381          DISP ,SPREAD;"MONTHS"
390          DISP "PLEASE TYPE IN THE NEW SPREAD"
```

```
400          INPUT WSPREAD
410          IF WSPREAD[illegible]2 OR WSPREAD[illegible]24 THEN GOTO ERR4
411 NEWSPR:  DISP "THE NEW MOVING AVERAGE PERIOD IS:"
412          DISP ,WSPREAD;"MONTHS
413          DISP "IS THIS CORRECT(Y,N)"
414          INPUT RESP7$
415          IF RESP7$[illegible]"Y" AND RESP7$[illegible]"N" THEN GOTO NEWSPR
416          IF RESP7$="N" THEN GOTO MOVAVE
417          SPREAD=WSPREAD
420          PRINT# 9,1 : CONTLIM,MONTH,YEAR,SPREAD
430          RETURN
440 ERR1:  DISP "THE CONTROL LIMIT MUST BE BETWEEN 1 AND 3 SIGMA"
450        GOTO CNTLIM
460 ERR2:  DISP "THE YEAR MUST BE GREATER THAN 1980 AND LESS THAN 2000"
470        GOTO DATCHG
480 ERR3:  DISP "THE MONTH MUST BE BETWEEN 1 AND 12"
490        GOTO DATCHG
500 ERR4:  DISP "THE MOVING AVERAGE PERIOD MUST BE BETWEEN 2 AND 24 MONTHS"
510        GOTO MOVAVE
540 REM  ************  PROGRAM CONTMAINT TERMINATE  ********************
```

```
10   REM  ***********  PROGRAM I  MAINT  ************************
20   REM  **  THIS PROGRAM IS USED TO CHANGE THE DEFAULT  **
30   REM  **  VALUES FOR INCLUDING TYPES WHEN:            **
40   REM  **   1.PROCESSING DEPOT TOTALS                  **
50   REM  **   2.PROCESSING BRANCH TOTALS                 **
60   REM  **   3.PROCESSING A NUMBER OF INDIVIDUAL TYPES  **
70   REM  **  WHICH WERE INITIALLY SET WHEB THE TYPE WAS  **
80   REM  **  LOADED.                                      **
90   REM  ***********************************************
100         OPTION BASE 1
110 DIM EXCINC$[3]
161 CLEAR
162         DISP
163         DISP ,"*** DEFAULT MAINTENANCE FUNCTION ***"
164         DISP
165         DISP
170 BRNCH: DISP "WHICH BRANCH"
180          INPUT BRANCH
190          DISP "WHICH DEPOT"
200          INPUT DEPOT
210  FLTYP: DISP "WHICH TYPE "
220          DISP "(TYPE CODE OR 0 TO END)"
230          INPUT TYPE
235          IF TYPE 1 OR TYPE 100 THEN DISP "TYPE INVALID" @ GOTO FLT
240         IF TYPE=0 THEN GOTO FINI  ELSE NEWFL
250 NEWFL: DISP ," OU ARE INTERESTE  IN
260         I S    BRANCH  BRANCH:" DEPOT ";DEPOT;" TYPE ";TYPE
270         DISP ,'IS THIS CORRECT(Y N)"
280         INPUT VALID1$
290         IF VALID1$="N" THEN GOTO BRNCH
291         IF VALID1$    N" AND VA I  $          THEN GOTO NEWFL
292         BRANCH$  AL$ (BRANCH)
293         D    T$ AL$ (DEPOT)
299 READFL: ON ERROR GOTO ERR1
300          ASSIGN# 1 TO  BRANCH$ I    T$ "INX:D    "
301          OFF ERROR
320          READ# 1.TYPE : POSITION.EXCINC$.FUEL$
330          IF P  ITION=0 THEN DI     THIS TYPE DOES NOT EXIST" @ GOTO
340 DSPXST: DI   THE EXISTING DEFAULTS ARE:"
350          DISP
360          DI
370          DISP "INCLUDE TYPE IN DEPOT  TOTAL       ".EXCINC$[2.2]
380          DISP "INCLUDE TYPE IN BRANCH TOTALS      ".EXCINC$[3.3]
390          DISP "                        FUEL CLASS =".FUEL$
400          DISP "DO YOU WANT TO CHANGE THESE DEFAULTS(Y N)"
410          INPUT RESP4$
420          IF RESP4$    "Y" AND RESP4$    "N" THEN GOTO DSPXST
430          IF RESP4$="N  THEN GOTO ADDCHG
1340 SETEXCINC: CLEAR
1341 DISP
1342 DISP
1350 IN1: DISP ,"MUST THIS TYPE BE INCLUDED WHEN PROCESSING"
1360             DISP ,"A NO OF INDIVIDUAL TYPES(Y N)"
1370             INPUT EXIN1$
1371             IF EXIN1$    "Y" AND EXIN1$    "N" THEN GOTO IN1
1380 IN2: DISP ,"MUST THIS TYPE BE INCLUDED WHEN PROCESSING"
1390             DISP ,'DEPOT TOTALS(Y,N)"
1400             INPUT EXIN2$
1401             IF EXIN2$    "Y" AND EXIN2$    "N" THEN GOTO IN2
1410 IN3: DISP ,"MUST THIS TYPE BE INCLUDED WHEN PROCESSING"
14           DISP ,"BRANCH TOTALS(Y N)"
1420             INPUT EXIN3$
```

```
1431           IF EXIN3$    " " AND EXIN $    "N" THEN GOTO IN3
1440           EXCINC$=EXIN1$&E IN
1450 [illegible]
1460           INPUT FUEL$
1461           IF FUEL$    "F" AND FUEL$    "D" THEN GOTO SETFUEL
1470 DISPINF : DISP "YOU HAVE SET THE FOLLOWING DEFAULTS:"
1480         DISP
1490         DISP "INCLUDE TYPE IN SERIES OF TYPES=",EXIN1$
1500         DISP "INCLUDE TYDE IN DEPOT  TOTALS  =",EXIN2$
1510         DISP "INCLUDE TYPE IN BRANCH TOTALS  =",EXIN $
1520         DISP "                         FUEL CLASS=",FUEL$
1530         DISP
1540         DISP "IS THIS CORRECT"
1550         INPUT RESP5$
1560         IF RESP5$    "Y" AND RESP5$    "N" THEN GOTO DISPINF
1570         IF RESP5$="N" THEN GOTO SETEXCINC
1580 WRITFL: PRINT# 1,TYPE ; POSITION,EXCINC$,FUEL$
1590 ADDCHG: DISP "DO YOU WISH TO MAKE FURTHER CHANGES(Y N)"
1591        INPUT RESP6$
1592        IF RESP6$    "Y" AND RESP6$    "N" THEN GOTO ADDCHG
1593        IF RESP6$="Y" THEN GOTO BRNCH  ELSE FINI
1600 FINI: ASSIGN# 1 TO  *
1605       DISP ,"PROCESSING COMPLETE'
1610       CHAIN "MAINMENU"
1620       STOP
1630 ERR1: DISP "THIS DEPOT-BRANCH COMBINATION DOES NOT EXIST"
1640       OFF ERROR @ GOTO BRNCH
1730 REM  ************  PROGRAM DEFMAINT TERMINATE  ********************
```

```
10 REM       ******************  PROGRAM  BRANMAINT  ******************
20 REM       **  THIS PROGRAM IS USED TO LOAD ,CHANGE AND DELETE      **
30 REM       **  DESCRIPTIONS OF BRANCHES IN ORDER THAT THE  MAY      **
40 REM       **  BE PRINTED FOR THE CORRECT CODES. ANY NAMES ARE      **
50 REM       **  ALLOWED , BOTH ALPHA AND NUMERIC. HOWEVER THE        **
60 REM       **  DESCRIPTION MAY NOT BE LONGER THAN 30 CHARACTERS    **
70 REM       **********************************************************
80 OPTION BASE 1
90 CLEAR
91  DISP
92  DISP ,"***  BRANCH MAINTENANCE FUNCTION ***"
93  DISP
110 ASSIGN# 7 TO  "BRANMAS:D700"
120 CODREQ: DISP "WHAT IS THE CODE OF THE BRANCH YOU WISH TO LOAD"
130            INPUT BRAN
135            IF BRAN 1 OR BRAN 50 THEN DISP ,"ONLY BRANCHES 1 TO 50 ARE ALLOWED"
 @ GOTO CODREQ
140            READ# 7,BRAN ; BRNDESC$,MEAN,RANGE
150            DISP "THE EXISTING DESCRIPTION FOR THIS CODE IS"
160            DISP BRNDESC$
170            DISP
180            DISP "PLEASE TYPE IN THE NEW DESCRIPTION"
190            INPUT BRNDESC$
200            DISP "THE NEW DESCRIPTION IS NOW"
210            DISP BRNDESC$
220            DISP "CORRECT (Y/N)"
230            INPUT RESP$
240            IF RESP$    "Y" THEN GOTO CODREQ
250            PRINT# 7,BRAN ; BRNDESC$,MEAN,RANGE
270            ASSIGN# 7 TO  *
280            DISP ,"PROCESSING COMPLETE"
285            CHAIN "MAINMENU"
290            STOP
300  REM   *****************    PROGRAM BRANMAINT TERMINATE    ***************
```

```
10 REM      *** ********.******  PROGRAM  DEPOTMAINT  ****************
20 REM      **  THIS PROGRAM IS USED TO LOAD ,CHANGE AND DELETE     **
30 REM      **  DESCRIPTIONS OF DEPOTS IN ORDER THAT THEY MAY       **
40 REM      **  BE PRINTED FOR THE CORRECT CODES. ANY NAMES ARE     **
50 REM      **  ALLOWED . BOTH ALPHA AND NUMERIC. HOWEVER THE       **
60 REM      **  DESCRIPTION MAY NOT BE LONGER THAN 30 CHARACTERS    **
70 REM      ********************************************************
80 OPTION BASE 1
90 CLEAR
    DISP
    DISP ,"***  DEPOT MAINTENANCE FUNCTION ***"
    DISP
110 ASSIGN# 8 TO  "DEPMAS:D700"
120 CODRED: DISP "WHAT IS THE CODE OF THE DEPOT YOU WISH TO LOAD"
130            INPUT DEP
135            IF DEP 1 OR DEP 99 THEN DISP ,"ONLY DEPOTS 1 TO 99 ARE ALLOWED  G
OTO CODRED
140            READ# 8,DEP ; DEPDESC$
150            DISP "THE EXISTING DESCRIPTION FOR THIS CODE IS"
160            DISP DEPDESC$
170            DISP
180            DISP "PLEASE TYPE IN THE NEW DESCRIPTION"
190            INPUT DEPDESC$
200            DISP "THE NEW DESCRIPTION IS NOW"
210            DISP DEPDESC$
220            DISP "CORRECT (Y/N)"
230            INPUT RESP$
240            IF RESP$ <  "Y" THEN GOTO CODRED
250            PRINT# 8,DEP ; DEPDESC$
270            ASSIGN# 8 TO  *
280            DISP ,"PROCESSING COMPLETE
               CHAIN "MAINMENU"
290            END
300   REM  ************   PROGRAM DEPOTMAINT TERMINATE   ****************
```

```
10 REM   ****************   PROGRAM STATTABMNT   ********************
20 REM   *** THIS PROGRAM IS USED TO MAKE CHANGES TO THE TABLE    ***
30 REM   *** OF STATISTICAL FACTORS USED IN THE CALCULATION OF    ***
40 REM   *** OF THE MEAN AND RANGE CONTROL CHARTS                 ***
50 REM   ***********************************************************
60 CLEAR
70 OPTION BASE 1
90 ASSIGN# 5 TO  "STATTABLE:D700"
160 DISP ,"***  STATISTICS TABLE MAINTENANCE ***"
170 DISP
180 DISP
190 INPUT1: DISP "PLEASE ENTER THE GROUPSIZE"
200          INPUT GRPSIZ
210          IF GRPSIZ 1 OR GRPSIZ 2  THEN GOTO INPUT1
220 READ# 5,GRPSIZ ; XFACT,RLLIM,RULIM
230 DISP "THE CURRENT FACTORS ARE SET TO:"
240 DISP
250 DISP "XFACTOR        =";XFACT
260 DISP "LOWER RFACTOR =";RLLIM;"          GROUPSIZE =";GRPSIZ
270 DISP "UPPER RFACTOR =";RULIM
280 DISP
290 INPUT2: DISP "ARE YOU SURE YOU WANT TO CHANGE THESE FACTORS(Y/N)"
300          INPUT RESP2$
310          IF RESP2$    "Y" AND RESP2$    "N" THEN GOTO INPUT2
320          IF RESP2$="N" THEN GOTO FINI
330 DISP "PLEASE TYPE IN THE NEW XFACTOR"
340 INPUT XFACT
350 DISP "PLEASE TYPE IN THE NEW LOWER RFACTOR"
360 INPUT RLLIM
370 DISP "PLEASE TYPE IN THE NEW UPPER RFACTOR"
380 INPUT RULIM
390 DISP "THE FACTORS YOU HAVE LOADED ARE:"
400 DISP
410 DISP "XFACTOR        =";XFACT
420 DISP "LOWER RFACTOR =";RLLIM;"             GROUPSIZE =";GRPSIZ
430 DISP "UPPER RFACTOR =";RULIM
440 DISP
450 DISP "IS THIS CORRECT(Y N)"
460 INPUT3: INPUT RESP3$
470          IF RESP3$    "Y" AND RESP3$    "N" THEN GOTO INPUT3
480          IF RESP3$="N" THEN GOTO INPUT2
490 PRINT# 5,GRPSIZ ; XFACT,RLLIM,RULIM
500 INPUT4: DISP "DO YOU WISH TO CHANGE ANY OTHER FACTORS(Y/N)"
510          INPUT RESP4$
520          IF RESP4$    "Y" AND RESP4$    "N" THEN GOTO INPUT4
530          IF RESP4$="Y" THEN GOTO INPUT1
540 FINI: ASSIGN# 5 TO  *
550        DISP ,"PROCESSING COMPLETE"
555        CHAIN "MAINMENU"
560        STOP
570 REM   ************  PROGRAM  STATTABMNT COMPLETED  *****************
```

```
10 REM    ****************   PROGRAM PARAMMNT  ************************
20 REM    **  THIS PROGRAM MAINTAINS THE MEAN AND RANGE OF EACH        **
30 REM    **  TYPE WITHIN ITS BRANCH-DEPOT COMBINATION. IT ALSO        **
40 REM    **  MAINTAINS THE DEPOT MEAN AND RANGE.                      **
50 REM    **  IT IS IMPORTANT TO NOTE THAT THERE IS ONE MEAN-RNG       **
60 REM    **  FILE FOR EACH DEPOT-BRANCH COMBINATION.                  **
70 REM    **  ALSO NOTE THAT BRANCH MEANS AND RANGES ARE HELD ON A     **
80 REM    **  SEPARATE FILE (BRANCH DESCRIPTION FILE                   **
90 REM    ***************************************************************
100 OPTION BASE 1
101 INITL: CLEAR
105        DISP ,"***  PARAMETER MAINTENANCE ***"
106        DISP
110        DISP ,"YOU MAY MAKE THE FOLLOWING CHANGES"
1[illegible]        DISP
130        DISP ,"1.CHANGE MEAN/RANGE OF A TYPE"
140        DISP
150        DISP ,"2.CHANGE MEAN/RANGE OF A DEPOT"
160        DISP
170        DISP ,"3.CHANGE MEAN/RANGE OF A BRANCH"
180        DISP
181        DISP ,"4.END"
182        DISP
190        DISP "MAKE YOUR SELECTION"
200        INPUT RESP1
210        IF RESP1 1 OR RESP1 4 THEN GOTO LDBRN
220        ON RESP1 GOSUB TYPCHG ,DEPCHG ,BRNCHG ,FINI
221        GOTO INITL
230 TYPCHG: GOSUB BRNENO
240         GOSUB DEPENO
250         GOSUB TYPENO
270 READ# 1,TYPE ; POSITION,EXCINC$,FUEL$
280 IF POSITION  0 THEN GOTO LDTYP
290 DISP "THERE EXISTS NO DATA FOR THIS TYPE..PLEASE LOAD DATA FIRST"
300 WAIT 1000 @ GOTO FINI
310 LDTYP: READ# 4,POSITION ; IND,MEAN,RANGE
320 DISP "THE EXISTING VALUES ARE:"
330 DISP "               MEAN =",MEAN
340 DISP "               RANGE=",RANGE
350 DISP "PLEASE TYPE IN THE NEW VALUES IN FORMAT MEAN,RANGE"
360 INPUT MEAN,RANGE
361 CATG=1
362 GOSUB NEWVAL
365 IND=1
370 PRINT# 4,POSITION ; IND,MEAN,RANGE
380 RETURN
390 DEPCHG: GOSUB BRNENO
400         GOSUB DEPENO
420 READ# 1,100 ; POSITION,EXCINC$,FUEL$
430 IF POSITION  0 THEN GOTO LDDEP
440 DISP "THERE EXIST NO DATA FOR THIS DEPOT....PLEASE LOAD DATA FIRST"
445 WAIT 1000 @ GOTO FINI
450 LDDEP: READ# 4,POSITION ; IND,MEAN,RANGE
460 DISP "THE EXISTING VALUES ARE:"
470 DISP "               MEAN =",MEAN
480 DISP "               RANGE=",RANGE
490 DISP "PLEASE TYPE IN THE NEW VALUES IN FORMAT MEAN,RANGE"
500 INPUT MEAN,RANGE
501 CATG=2
502 GOSUB NEWVAL
505 IND=1
510 PRINT# 4,POSITION ; IND,MEAN,RANGE
520 RETURN
```

```
530 BRNCHG: GOSUB BRNENQ
531 ASSIGN# 7 TO "BRANM[illegible]:D700"
532 ON ERROR GOTO ERR2
540 LDBRN: READ# 7,BRANCH ; BRNDESC$,MEAN,RANGE
541 OFF ERROR
550 DISP "THE EXISTING VALUES ARE:
560 DISP "              MEAN= ",MEAN
570 DISP "              RANGE= ",RANGE
580 DISP "PLEASE TYPE IN THE NEW VALUES IN FORMAT MEAN,RANGE"
590 INPUT MEAN,RANGE
591 CATG=3
592 GOSUB NEWVAL
600 PRINT# 7,BRANCH ; BRNDESC$,MEAN,RANGE
610 RETURN
620 BRNENQ: DISP "WHICH BRANCH"
630         INPUT BRANCH
640         BRANCH$=VAL$ (BRANCH)
650         RETURN
660 DEPENQ: DISP "WHICH DEPOT"
670         INPUT DEPOT
680         DEPOT$=VAL$ (DEPOT)
710 NEWFL: DISP "IS THIS A NEW FILE(Y/N)"
720        INPUT RESP2$
730        IF RESP2$ [illegible] "Y" AND RESP2$ [illegible] "N" THEN GOTO NEWFL
740        IF RESP2$="N" THEN GOTO EXISTFL
745        ON ERROR GOTO ERR4
750        CREATE BRANCH$&DEPOT$&"PAR:D700",15,24
751        OFF ERROR
752        ON ERROR GOTO ERR1
758        ASSIGN# 1 TO BRANCH$&DEPOT$&"INX:D700"
760        ASSIGN# 4 TO BRANCH$&DEPOT$&"PAR:D700"
761        MEAN=0
762        RANGE=0
763        IND=0
764        OFF ERROR
770        FOR J=1 TO 15
780        PRINT# 4,J ; IND,MEAN,RANGE
790        NEXT J
800        RETURN
809 EXISTFL: ON ERROR GOTO ERR3
810          ASSIGN# 4 TO BRANCH$&DEPOT$&"PAR:D700"
811          ASSIGN# 1 TO BRANCH$&DEPOT$&"INX:D700"
812          OFF ERROR
820          RETURN
830 TYPENQ: DISP "WHICH TYPE"
840         INPUT TYPE
845         IF TYPE [illegible] OR TYPE [illegible] THEN DISP "TYPE INVALID" @ GOTO TYPENQ
850         RETURN
860 ERR1: DISP "THERE EXISTS NO DATA FOR THIS FILE-PLEASE LOAD DATA FIRST"
870       WAIT 1000 @ GOTO FINI
880 ERR2: DISP "THIS BRANCH IS NOT ON THE BRANCHFILE - PLEASE LOAD IT FIRST"
890       WAIT 1000 @ GOTO FINI
900 ERR3: DISP "THIS FILE DOES NOT EXIST" @ OFF ERROR @ GOTO NEWFL
901 ERR4: DISP "THIS FILE ALREADY EXISTS" @ OFF ERROR @ GOTO NEWFL
910 FINI: DISP ,"PROCESSING COMPLETE"
911       CHAIN "MAINMENU"
915       STOP
920 NEWVAL: DISP "THE NEW VALUES YOU HAVE LOADED ARE:"
930         DISP
940         DISP "              MEAN = ",MEAN
950         DISP "              RANGE= ",RANGE
960         DISP "IS THIS CORRECT(Y/N)"
970         INPUT RESP9$
980         IF RESP9$ [illegible] "Y" AND RESP9$ [illegible] "N" THEN GOTO NEWVAL
990         IF RESP9$="Y" THEN RETURN
1000        ON CATG GOTO LDTYP ,LDDEP ,LDBRN
1020 REM ************ PROGRAM PARAMNT TERMINATE ***************
```

```
10   REM  ***********    PROGRAM   SIMPLE   ***************
20   REM  **  THIS PROGRAM CALCULATES THE SIMPLE DATA PLOT      **
30   REM  **  GRAPHS  FOR A VEHICLE TYPE OR GROUP OF VEHICLE    **
40   REM  **  TYPES OR AN ENTIRE DEPOT.                         **
50   REM  **  THE DEFAULT IS TAKEN FROM THE INDEX FILE FOR      **
60   REM  **  EACH TYPE IF IT IS NOT ENTERED.                   **
70   REM  **  IT THEN WRITES THE DATA TO A FILE FOR LATER       **
80   REM  **   PLOTTING.                                        **
90   REM  ******************************************************
100   CLEAR
110 OPTION BASE 1
111  DISP
112  DISP ,"*** PLOT SIMPLE DATA GRAPHS ***"
113  DISP
120  DISP ,"PROCESSING  BEGINS"
130  DISP ,"VARIABLE INITIALIZATION"
140  DIM TOTAL(36,3),TYPES(100),STORETOT(36)
150  COM TYPTOT(36,51),GRAPHTYP
160  COM DEPOT(10),BRANCH
170  COM TYPETAB(51),FUELTAB$(51)[1]
180       FOR J=1 TO 36
190           FOR L=1 TO 3
200               TOTAL(J,L)=0
210           NEXT L
250           FOR L=1 TO 51
260               TYPTOT(J,L)=0
270            NEXT L
280       NEXT J
290       FOR J=1 TO 51
300           TYPES(J)=0
310       NEXT J
311       FOR J=1 TO 36
312           STORETOT(J)=0
313       NEXT J
320  ASSIGN# 9 TO  "CONTMAS:D700"
330  READ# 9,1 ; CONTLIM,MONTH,YEAR,SPREAD
335 GRAPHDEF: DISP
340              DISP ,"WHICH OF THE FOLLOWING DO YOU WISH TO DO"
350            DISP ,"1.PRINT THE GRAPH FOR A SERIES OF TYPES WITHIN A DEPOT"
360            DISP
370            DISP ,"2.PRINT THE TOTAL GRAPH FOR THE DEPOT"
380            DISP
390            DISP ,"3.PRINT THE TOTAL GRAPH FOR THE BRANCH"
400            INPUT GRAPHTYP
410            IF GRAPHTYP>3 OR GRAPHTYP<1 THEN GOTO GRAPHDEF
420 DISBRN: DISP "WHICH BRANCH"
430            INPUT BRANCH
440            LET L=1
450  DEPIN: DISP "WHICH DEPOTS TO INCLUDE"
460            DISP " (TYPE CODE OR 0 TO END)"
470            INPUT DEPOT(L)
480            IF DEPOT(L)=0 THEN GOTO FINI  ELSE ADDDEP
490 NEXTDEP: DISP "MORE DEPOTS"
500             DISP "(TYPE CODE OR 0 TO CONTINUE)"
510             INPUT DEPOT(L)
520             IF DEPOT(L)=0 THEN GOTO DEPFIN  ELSE ADDDEP
525  ADDDEP: IF GRAPHTYP=3 THEN DEPOT(L+1)=0 @ GOTO DEPFIN
530             L=L+1
535            IF L>10 THEN DISP "NO OF DEPOTS EXCEEDED" @ DEPOT(L)=0 @ GOTO DEPFIN

540             GOTO NEXTDEP
550  DEPFIN: LET J=1
```

```
[illegible]
1150 FUELCT: [illegible]
[illegible]
1200 [illegible]
1210         IF [illegible] ELSE RETIE [illegible]
1220 [illegible]
1230 FUELCH: DISP "[illegible] DIESEL OR PETROL (D/P)"
1240         INPUT FUELCH$
1250         IF FUELCH$ <> "D" AND FUELCH$ <> "P" THEN GOTO FUEL[illegible]
1260         RETURN
1270 ERR1: DISP , "THIS [illegible] COMBINATION DOES NOT [illegible]"
1280         GOTO [illegible]
1440 PE "************** PROGRAM [illegible] **************"
```

```
10 REM ********************* PROGRAM SUBDATPLT *********************
20 REM *** THIS PROGRAM IS CALLED BY PROGRAM SUMDAT.             ***
30 REM *** IT PLOTS A GRAPH OF THE ACTUAL CONSUMPTION DATA OF    ***
40 REM *** A BRANCH, A DEPOT, OR A SERIES OF TYPES AFTER EACH OF ***
50 REM *** WHICH IT PROMPTS FOR THE PLOTTER TO BE SET UP FOR THE ***
60 REM *** FOLLOWING ONE.                                         ***
70 REM ***********************************************************
80  OPTION BASE 1
81  DISP
82  DISP "*** SIMPLE DATA PLOTS ***"
90  COM TYPDOT(12,51),BRANNT(B
100 COM DEPOT(40),BRANCH
110 DIM TYPETAB(51),FUELRAST(5,12)
112 ON ERROR GOSUB ERR1
115 PLOTTER IS [illegible]
120 OFF ERROR
130 LET J=0
140 ASSIGN# 7 TO "CONTRAS:D700"
150 READ# 7,1 ; [illegible]
155 CONTROL[illegible]
160 LET L=1
170 LOCATE 15,115,15,75
180 ASSIGN# 7 TO "BRANAME:D700"
190 ASSIGN# 8 TO "DEPNAM:D700"
200 READBRN: READ# 7,BRANCH ; BRNDESC$
210           IF BRANNT(B[illegible]) THEN GOTO READDEP
220           DEPDESC$="ALL"
230 SETUP: DISP "PLEASE SET UP YOUR PAPER & PRESS C TO CONT OR E TO END"
240        INPUT RESP$
250        IF RESP$="E" THEN GOTO FIN1
260        IF RESP$="C" THEN GOTO BEGIN ELSE SETUP
270 BEGIN: YMAX=0
280        YMIN=99999
290 SETYMINMAX: FOR I=1 TO 12
300            [illegible]
310            [illegible]
320            NEXT I
330            [illegible]
335            [illegible]
338            [illegible]
340            [illegible]
350            [illegible]
360            [illegible]
370            [illegible]
380            [illegible]
390            [illegible]
400        [illegible]
410        [illegible]
420        FRAME
430        [illegible]
440        [illegible]
450        [illegible]
460        [illegible]
470        [illegible]
475        [illegible]
480        FOR J=1 TO 12
490        DRAW J,TYPDOT(J,[illegible])
500        NEXT J
510        [illegible]
515        [illegible]
520        [illegible]
```

```
530          CSIZE 4
540 LABELL:  LABEL USING FORM1 ; "[illegible]"
550          MOVE [illegible]
560          CSIZE 3
560          IF CONTMONTH [illegible]
ONE ,MONIO ,[illegible]
570          LABEL USING FORM1 ; CONTMONTH$;[illegible]
580          CSIZE 3
590          MOVE 40,85
600          LABEL USING FORM2 ; [illegible]
610          MOVE 40,80
612          [illegible]
620          LABEL USING FORM2 ; "[illegible]"
630          CSIZE 4
640          MOVE 12,75
650          IF GRAPHTYP=1 THEN GOTO VERTAXIS
660          IF GRAPHTYP=1 THEN GOTO [illegible]
661          LABEL USING "[illegible]" ; "VEHICLE" ,"TYPE"
681          FOR M=1 TO 15
682          IF TYPETAB(M)=[illegible] THEN GOTO LABTYP
683          NEXT M
684          LABEL USING "K" ; "VARIOUS" @ GOTO VERTAXIS
670 LABTYP:  FOR M=1 TO 15
680          IF TYPETAB(M)=[illegible] THEN GOTO VERTAXIS
690          LABEL USING "K" ; TYPE$(M)
700          NEXT M
710 FORMA: IMAGE 32A
720 FORMB: IMAGE 8A,X,[illegible]
730 FORMC: IMAGE 4A,DDDD
735 FORMD: IMAGE 8A,X
740 VERTAXIS: MOVE 7,8
750             CSIZE 3
760             LABEL USING "[illegible]" ; "NUMBER" ,"[illegible]"
770             MOVE 1,7,1
780             LABEL USING "[illegible]" ; "MONTHS"
790             SETUU
800 IF GRAPHTYP=1 THEN LAL=1 ELSE GOTO FINI
810 IF LAST OF TYPETAB(I+1)=[illegible] THEN GOTO FINI
820 GOTO BEGIN
830 FINI: DISP "PROCESSING COMPLETE"
835          [illegible]
840 PEN 0
850 STOP
860 READDEF: [illegible]
870          GOTO BEGIN
880 [illegible] I=I+1
890          LABEL USING "[illegible]" ; "VEHICLE" ,"TYPE" ,TYPETAB(I)
900          GOTO VERTAXIS
910 LABAXIS: [illegible]
920          FOR M=1 TO 12 STEP 1
930          MOVE [illegible]
931          [illegible]
940          IF MONTH=12 THEN [illegible]
950          IF [illegible] THEN [illegible]
960          IF [illegible] THEN [illegible]
970          ON [illegible] ,[illegible]
[illegible] ,[illegible] ,[illegible]
980          LABEL MONTHS[illegible]
1000          NEXT M
1010          SETUU
1020          MOVE [illegible]
1030          LABEL [illegible]
1040          SETUU
1050          [illegible]
1060          FOR [illegible] TO [illegible]
1070          MOVE [illegible]
1080          [illegible]
[illegible]
```

```
1090        RETURN
1100 MON1: MONTHDESC$="JAN"
1110        RETURN
1120 MON2: MONTHDESC$="FEB"
1130        RETURN
1140 MON3: MONTHDESC$="MAR"
1150        RETURN
1160 MON4: MONTHDESC$="APR"
1170        RETURN
1180 MON5: MONTHDESC$="MAY"
1190        RETURN
1200 MON6: MONTHDESC$="JUN"
1210        RETURN
1220 MON7: MONTHDESC$="JUL"
1230        RETURN
1240 MON8: MONTHDESC$="AUG"
1250        RETURN
1260 MON9: MONTHDESC$="SEPT"
1270        RETURN
1280 MON10: MONTHDESC$="OCT"
1290        RETURN
1300 MON11: MONTHDESC$="NOV"
1310        RETURN
1320 MON12: MONTHDESC$="DEC"
1330        RETURN
1340 FUELRTN: IF FUELTAB$(I+1)="P" THEN FUELDESC$="PETROL" & RETURN
1350        IF FUELTAB$(I+1)="D" THEN FUELDESC$="DIESEL" ELSE FUELDESC$="
1360        RETURN
1370 EXIT: DISP "PLEASE SWITCH THE PLOTTER OFF"
1380        DISP "CONT(Y)"
1390        INPUT PLOTTER$
1400        IF PLOTTER$<>"Y" THEN RET
1410        GOTO XINI
1420 REM ***************** PROGRAM IMPS-TALT COMPLETE *****************
```

```
10   REM  ****************   PROGRAM     .IFAVE   ****************
20   REM  **  THIS PROGRAM CALCULATES THE MOVING AVERAGE          **
30   REM  **  FIGURES FOR A VEHICLE TYPE OR GROUP OF VEHICLE      **
40   REM  **  TYPES OR AN ENTIRE DEPO                             **
50   REM  **  THE DEFAULT IS TAKEN FF        INDEX FILE FOR       **
60   REM  **  EACH TYPE IF IT IS NOT ENTERED.                     **
70   REM  **  IT THEN WRITES THE DATA TO   FILE FOR LATER         **
80   REM  **  PRINTING OR PLOTTING.                               **
90   REM  ************************************************************
100   CLEAR
110 OPTION BASE 1
111  DISP
112  DISP ,"*** MOVING AVERAGE GRAPHS ***"
113  DISP
120  DISP ,"  PROCESSING  BEGINS"
130  DISP ,"  VARIABLE INITIALIZATION"
140  DIM TOTAL(36.3 .TYPES(100) STOREAVE(36.2)
150  COM TOTAVE(36.51).GRAPHTYP
160  COM DEPOT 10).BRANCH
170  COM TYPETAB(51).FUELTAB$(51)[1]
180       FOR J-1 TO 36
220           FOR L=1 TO .
230               STOREAVE(J.L)=0
240           NEXT L
250           FOR L=1 TO 51
260               TOTAVE(J.L)=0
270            NEXT L
280       NEXT J
290       FOR J=1 TO 51
300           TYPES(J)=0
310       NEXT J
320  ASSIGN# 9 TO  "CONTMAS:D700"
330  READ# 9.1 ; CONTLIM.MONTH.YEAR.SPEE )
340 GRAPHDEF: DISP
341          DISP ,"YOU MAY DO ANY OF THE FOLLOWING:"
             DISP ,"1.PRINT THE GRAPHS FOR A SERIES OF TYPES WITHIN A DEPOT"
             DI
370          DISP ,"2.PRINT THE TOTAL GRAPH FOR THE DEPOT"
380          DISP
390          DISP ,"3.PRINT THE TOTAL GRAPH FOR THE BRANCH"
391          DISP "MAKE YOUR SELECTION"
400          INPUT GRAPHTYP
41           IF GRAPHTYP 3 OR GRAPHTYP 1 THEN GOTO GRAPHDEF
420  B NCH: DISP "WHICH BRANCH"
430  INPUT BRANCH
440  LET L=1
45   DEPIN: DISP "WHICH DEPOTS TO INCLUDE
460          DISP " (TYPE CODE OR   TO END)"
470          INPUT DEPOT(L)
480          IF DEPOT(L)=0 THEN CHAIN "MAINMENU" ELSE ADDDEP
490 NEXTDEP: DISP "MORE DEPOTS"
500           DISP "(TYPE CODE OR 0 TO CONTINUE)"
510           INPUT DEPOT(L)
520           IF DEPOT(L)=0 THEN GOTO DEPIN  ELSE ADDDEP
529  ADDDEP: IF GRAPHTYP   THEN DEPOT(L+1)=0 @ GOTO DEPFIN
             L=L+1
             IF L=10 THEN DISP "NO OF DEPOTS EXCEEDED" @ DEPOT(L)=0 @ GOTO DEPFIN

540           GOTO NEXTDEP
550  DEPFIN: LET J=1
555           GOSUB DIEPET
559 TYPEIN: IF GRAPHTYP=. THEN TYP (J) 100 @ GOTO SETL
560           DISP "TYPES TO INCLUDE"
```

```
570          DISP "(TYPE CODE OR 0 TO END OR 100 FOR DEFAULT)"
580          INPUT TYPES(J)
585          IF TYPES(J) 0 OR TYPES(J) 100 THEN DISP "TYPE INVALID" @ GOTO TYPEIN

590          IF TYPES(J)=100 THEN GOTO SETL
600          IF TYPES(J)=0 THEN CHAIN "MAINMENU" ELSE ADDTYP
610 NEXTYPE: DISP "MORE TYPES"
620           DISP "(TYPE CODE OR 0 TO CONTINUE)"
630           INPUT TYPES(J)
631           IF TYPES(J) 0 OR TYPES(J) 100 THEN DISP "TYPE INVALID" @ GOTO NEXTY
PE
635           IF TYPES(J)-1 0 THEN DISP "YOU MAY NOT MIX SPECIFIC AND GENERAL REQ
UESTS" @ GOTO NEXTYPE
640           IF TYPES(J)=0 THEN GOTO SETL  ELSE ADDTYP
650 ADDTYP: J=J+1
651          IF J=51 THEN DISP "NO OF TYPES EXCEEDED" @ TYPES(J)=0 @ GOTO SETL
660          GOTO NEXTYPE
670 SETL: LET L=1
680         LET B=1
690          DISP ,"REQUEST PROCESSING"
691         BRANCH$=VAL$ (BRANCH
692 SETDE: DEPOT$(L)=VAL$ (DEPOT(L))
700 SETFL: DISP BRANCH$&DEPOT$(L)&"INX:D700"
705         ON ERROR GOTO ERR1
710         ASSIGN# 1 TO BRANCH$&DEPOT$(L)&"INX:D700"
720          ASSIGN# 2 TO BRANCH$&DEPOT$(L)&"DAT:D700"
725         OFF ERROR
730          LET K=1
740          IF TYPES(J)=100 THEN GOTO ALLREAD  ELSE READFL
750 ALLREAD: DISP ,"PROCESSING TYPE ",K
760           READ# 1,K ; POSITION,EXCINC$,FUEL$
765          IF POSITION=0 THEN DISP ,"TYPE",K,"DOES NOT EXIST" @ GOTO ADDK
770          IF FUELIN$   FUEL$ AND FUELIN$     B" THEN DISP "TYPE",K,"NOT INCLU
DED" @ GOTO ADDK
780          GOTO TYPELIM
790 TYPRET: READ# 2,POSITION ; IND,TOTAL(,)
800          GOTO TOTAL
810 READFL: DISP ,"PROCESSING TYPE",TYPES(K)
820          READ# 1,TYPES(K) ; POSITION,EXCINC$,FUEL$
830          IF POSITION#0 THEN GOTO READ2FL
840          DISP "TYPE ",TYPES(K),"DOES NOT EXIST"
850          GOTO ADDK
855 READ2FL: IF FUELIN$    FUEL$ AND FUELIN$     "B" THEN DISP "TYPE",TYPES(K),"N
OT INCLUDED" @ GOTO ADDK
860 READ2FL: READ# 2,POSITION ; IND,TOTAL(,)
870 TOTAL: GRANDTOT=0
880         FOR I=1 TO SPREAD
890          GRANDTOT=GRANDTOT+TOTAL(I,1)
900         NEXT I
910 AVRG: FOR I=SPREAD+1 TO  6
920          GRANDTOT=GRANDTOT+TOTAL(I,1)-TOTAL(I-SPREAD,1)
930          STORSAVE(I,1)=GRANDTOT/SPREAD
950          IF GRAPHTYP=1 THEN GOTO TYPAVE
960          TOTAL(I,1)=STORSAVE(I,1)+TOTAVE(I,1)
980 NEXTI: NEXT I
990  TYPSTR: IF GRAPHTYP   1 THEN FUELTAB$(1)=FUELIN$ ELSE FUELTAB$(B)=FUEL$
995           IF GRAPHTYP   THEN TYPETAB(1)=0 @ GOTO DISPTYP
1000          IF TYPES(J)      THEN TYPETAB(B)=K ELSE TYPETAB(B)=TYPES(K)
1001          TYPETAB(B+1)=0
1005          B=B+1
1010 DISPTYP: IF TYPES(J)=100 THEN DISP ,"COMPLETED TYPE ",K ELSE DISP ,"COMPLET
ED TYPE ",TYPES(K)
1020 ADDK: K=K+1
1030          IF K 100 THEN GOTO CHKTYP
1040          DISP ,"TYPE LIMIT REACHED
1050          GOTO ADDL
```

```
1060 CHKTYP: IF [illegible]
1070 [illegible]
1080 PRNT: [illegible] FOR I TO [illegible]
1090 [illegible]
1100 [illegible]
1110 CHAIN "[illegible] BELOT
1120 STOP
1130 TYPELIM: ON GRAPHTYP GOTO T[illegible] ,TYP[illegible] ,TYP
1140 TYP1: IF EXCINC$="NYY" OR EXCINC$="NYN" OR EXCINC$="NNY" OR EXCINC$="NNN" THEN DISP ,"TYPE EXCLUDED BY [illegible]
1150 TYP2: IF EXCINC$="YNY" OR EXCINC$="YNN" OR EXCINC$="NNY" OR EXCINC$="NNN" THEN DISP ,"TYPE EXCLUDED BY DEFAULT" @ GOTO ADDL ELSE GOTO TYPRET
1160 TYP3: IF EXCINC$="YYN" OR EXCINC$="YNN" OR EXCINC$="NYN" OR EXCINC$="NNN" THEN DISP ,"TYPE EXCLUDED BY DEFAULT" @ GOTO ADDL ELSE GOTO TYPRET
1170 TYPRET: TOT[illegible] ,D[illegible]DREA[illegible](I,I
1180 GOTO NEXTI
1200 ADDL: L=L+1
1210 IF DEPOT(L)=0 THEN GOTO [illegible]INI ELSE SETU[illegible]
1220 DIEPET: IF GRAPHTYP=1 THEN FUELIN$="B" @ RETURN
1300 FUEL[illegible]: DISP "DO YOU WISH TO PRINT DIE[illegible] OR [illegible]OL(D F)"
1310 INPUT FUELIN$
1320 IF FUELIN$ [illegible] "F" AND FUELIN$ [illegible] "D" THEN GOTO FUEL23
1330 RETURN
1340 ERR1: DISP . THIS BRANCH-DEPOT COMBINATION DOES NOT EXIST"
1350 GOTO BRNCH
1540 REM ************ PROGRAM M[illegible]INGAVE TERMINATE *********************
```

```
10 REM  ********************  PROGRAM MOVAVEPLOT  ********************
20 REM  ***  THIS PROGRAM IS CALLED BY PROGRAM MOVINGAVE             ***
30 REM  ***  IT PLOTS A GRAPH OF THE MOVING AVERAGE CONSUMPTION OF  ***
40 REM  ***  A BRANCH, A DEPOT, OR A RANGE OF TYPES AFTER EACH OF   ***
50 REM  *** WHICH IT PROMPTS FOR THE PLOTTER TO BE SET UP FOR THE   ***
60 REM  *** FOLLOWING ONE.                                          ***
70 REM  ***************************************************************
80  OPTION BASE 1
81  DISP
82  DISP ."*** MOVING AVERAGE PLOTS ***"
90  COM TOTAVE(24,21),GRAPHTYP
100 COM DEPOT(100),BRANCH
110 COM TYPETAB(51),FUELTAB(21)(21)
115 ON ERROR GOSUB ERR1
120 PLOTTER IS 805
125 OFF ERROR
130 LET I=0
140 ASSIGN# 9 TO "CONTMAX:D700"
150 READ# 9,1 ; CONTLIM,MONTH,YEAR,[illegible]
155 CONTMONTH=MONTH
160 LET L=1
170 LOCATE 15,115,15,75
180 ASSIGN# 7 TO "BRANNAME:D700"
190 ASSIGN# 8 TO "DEPNME:D700"
200 READBRN: READ# 7,BRANCH ; BRNDESC$
210          IF GRAPHTYP<>1 THEN GOTO READDEP
220          DEFDESC$="ALL"
230 SETUP: DISP "PLEASE SET UP YOUR PAPER & PRESS C TO CONT OR E TO END"
240         INPUT RESP$
250         IF RESP$="E" THEN GOTO FIN
260         IF RESP$="C" THEN GOTO BEGIN ELSE SETUP
270 BEGIN: YMAX=0
280        YMIN=99999
290 SETMINMAX: FOR K=1 TO 24
300             YMIN=MIN(YMIN,TOTAVE(K,L))
310             YMAX=MAX(YMAX,TOTAVE(K,L))
320             NEXT K
330             YSUM=YMAX-YMIN
335             IF YSUM>40000 THEN YDIFF=10000
336             IF YSUM<=40000 AND YSUM>20000 THEN YDIFF=2000
340             IF YSUM<=20000 AND YSUM>1000 THEN YDIFF=1000
350             IF YSUM<=1000 AND YSUM>500 THEN YDIFF=100
360             IF YSUM<=500 AND YSUM>100 THEN YDIFF=50
370             IF YSUM<=100 THEN YDIFF=25
380             YMIN=YMIN-DIFF
390             YMAX=YMAX+DIFF
400         SCALE 1,24,YMIN,YMAX
410         PEN 1
420         FRAME
430         PEN 0
440         AXES 1,YDIFF,1,YMIN
450         GOSUB LABELS
460         PEN 2
470         LINE TYPE 1
475         MOVE 1,TOTAVE(1,L)
480         FOR J=1 TO 24
490         DRAW J,TOTAVE(J,L)
500         NEXT J
510          SETDU
515         LORG 1
520         MOVE [illegible]
530         [illegible]
540 LABEL1: LABEL [illegible]
```

```
1090          RETURN
1100 MON1:    MONTHWORD$="JAN"
1110          RETURN
1120 MON2:    MONTHWORD$="FEB"
1130          RETURN
1140 MON3:    MONTHWORD$="MAR"
1150          RETURN
1160 MON4:    MONTHWORD$="APR"
1170          RETURN
1180 MON5:    MONTHWORD$="MAY"
1190          RETURN
1200 MON6:    MONTHWORD$="JUN"
1210          RETURN
1220 MON7:    MONTHWORD$="JUL"
1230          RETURN
1240 MON8:    MONTHWORD$="AUG"
1250          RETURN
1260 MON9:    MONTHWORD$="SEP"
1270          RETURN
1280 MON10:   MONTHWORD$="OCT"
1290          RETURN
1300 MON11:   MONTHWORD$="NOV"
1310          RETURN
1320 MON12:   MONTHWORD$="DEC"
1330          RETURN
1340 ERROR:   DISP "PLEASE [illegible] THE PLOTTER ON"
1350          DISP "[illegible]"
1360          INPUT PLOTTER$
1370          IF PLOTTER$<>"" THEN RETURN
1380          GOTO FINI
1390 FUELDESCR: IF FUELTYPE$[1,1]="P" THEN FUELDESCR$="PETROL" @ RETURN
1400            IF FUELTYPE$[1,1]="D" THEN FUELDESCR$="DIESEL" @ RETURN
1410            FUELDESCR$="[illegible]"
1420            RETURN
1430 REM ****************** PROGRAM "[illegible]PLOT" COMPLETE ******************
```

```
10  REM  ****************  PROGRAM CONTROL  ****************
20  REM  **  THIS PROGRAM READS THE [illegible]                    **
30  REM  **  OR GROUP OF [illegible] TYPES OR ENTIRE GROUP [illegible]  **
40  REM  **  [illegible] FOR EACH MONTH, THE MEAN AND THE [illegible]   **
50  REM  **  FROM THE FILE. IT ALSO CALCULATES THE [illegible]        **
60  REM  **  MEAN, RANGE AND [illegible] LIMITS FOR THE MEAN AND      **
70  REM  **  RANGE CHARTS.                                          **
80  REM  **  THESE ARE WRITTEN TO A FILE FOR LATER PLOTTING         **
90  REM  **                                                         **
100 REM  **********************************************************
110 CLEAR
111 DISP
112 DISP ,"*** CONTROL CHART FUNCTION ***"
113 DISP
120 OPTION BASE 1
130 DIM [illegible]
131 DIM [illegible]
132 DIM [illegible]
133 DIM [illegible]
134 DIM [illegible]
135 DIM [illegible]
140 [illegible]
141 DISP ,"    PROGRAM BEGINS"
142 DISP ,"[illegible]"
143     FOR N=1 TO [illegible]
144             [illegible]
145     NEXT N
150     FOR I=1 TO 24
160         FOR L=1 TO [illegible]
170             [illegible]
180         NEXT L
181         FOR L=1 TO [illegible]
182             [illegible]
183             [illegible]
184         NEXT L
185             [illegible]
186             [illegible]
187     NEXT I
190     [illegible]
191     [illegible]
210 [illegible]: DISP
211             DISP ,"YOU MAY DO ANY OF THE FOLLOWING:"
212             DISP
213             DISP ,"1) PRINT THE CHARTS FOR A SERIES OF TYPES WITHIN A GROUP"
214             DISP
215             DISP ,"2) PRINT THE [illegible] FOR THE GROUP"
216             DISP
217             DISP ,"3) PRINT THE [illegible] FOR THE [illegible]"
218             [illegible]
219             [illegible]
220             IF [illegible] THEN [illegible]
221 [illegible] DISP [illegible]
222 [illegible]
223 [illegible]
224 [illegible]
225 [illegible]
226 [illegible]
227 [illegible]
228 [illegible]
229 [illegible]
230 [illegible]
231 [illegible]
232 [illegible]
233 [illegible]
234 [illegible]
```

```
490          LCLD(M,K)=0
495          NEXT M
497          RETURN
500 CONTLIN: READ# 2,SUBPAR(1)[illegible] EXACT,LLIMR,ULIMR
520          LCL(M,K)=[illegible]MEAN(M)[illegible]*[illegible]RANGE(M)*CONTLIN/[illegible]
530          LCLX(M,K)=[illegible]MEAN(M)[illegible]*[illegible]RANGE(M)*CONTLIN/[illegible]
531          IF LCLX(M,K)<0 THEN LCLX(M,K)=0
540          UCLR(M,K)=ULIMR*[illegible]RANGE(M)*CONTLIN/[illegible]
550          LCLR(M,K)=LLIMR*[illegible]RANGE(M)*CONTLIN/[illegible]
551          IF LCLR(M,K)<0 THEN LCLR(M,K)=0
555          NEXT M
560          RETURN
565 READFL: DISP ,"PROCESSING TYPE",TYPES(I)
570         READ# 1,TYPES(I) ; POSITION,EXCING$,FUEL$
571          IF POSITION [illegible] THEN GOTO READ2FL
572          DISP "TYPE",TYPES(I),"DOES NOT EXIST"
573          GOTO NEXT1
590 READ2FL: IF FUELIN$ [illegible] FUEL$ AND FUELIN$ [illegible] "B" THEN DISP [illegible]
OT INCLUDED" @ GOTO ADD
591          READ# 2,POSITION ; IND,TOTAL(1)
600         RETURN
609 READALL: DISP ,"PROCESSING TYPE",K
610          READ# 1,K ; POSITION,EXCING$,FUEL$
620          IF POSITION=0 THEN DISP ,"TYPE",K,"DOES NOT EXIST" @ GOTO [illegible]
625          IF FUELIN$ [illegible] FUEL$ AND FUELIN$ [illegible] "B" THEN DISP [illegible]
UDED" @ GOTO ADD
630          GOTO TYPELIM
640 TYPPET: READ# 2,POSITION ; IND,TOTAL(1)
660          RETURN
670 [illegible]: FOR M=1 TO 24
675          IF TOTAL(M+12,2)=0 THEN SUBPAR[illegible]
680          MEAN(M,K)=TOTAL(M+12,1)
690          RANGE(M,K)=TOTAL(M+12,2)
700          SUBPAR1(M)=TOTAL(M+12,2)
701 NEXTM: NEXT M
710          RETURN
715 ACCUMTOT: FOR M=1 TO 24
720           TOTMEAN(M)=[illegible]MEAN(M)+TOTAL(M+12,1)
725           SUBPAR[illegible](M)=SUBPAR[illegible](M)+TOTAL(M[illegible])
730           IF SUBPAR[illegible](M)[illegible]21 THEN SUBPAR[illegible]
735 ADDM: NEXT M
738 [illegible]: FOR M=1 TO 24
740           [illegible](M)=TOTMEAN(M)[illegible]
742          NEXT M
748 RETURN
750 TOTMEAN: GOSUB [illegible]
751            FOR M=1 TO 24
752            MEAN(M,1)=TOTMEAN(M)
753            RANGE(M,1)=[illegible]
754            NEXT M
755      [illegible]
756      GOSUB [illegible]
757      GOTO [illegible]
761 [illegible]: IF [illegible]
[illegible]            [illegible]
[illegible]            RETURN
[illegible]: IF [illegible] THEN [illegible]
791          IF [illegible] THEN [illegible]
800          [illegible]
810          IF [illegible]
820 [illegible]
[illegible]
[illegible] DISP [illegible]
[illegible]
```

```
495            LCLR(M,K =0
496            NEXT M
497            RETURN
500 CONTLIM: READ# 5,SUBGRPSIZ(M) : XFACT,LLIMR,ULIMR
520            UCLX(M,K)=TMMEAN K +XFAC *TMRANGE K /*CONTLIM/3*SUBGRPSIZ M
530            LCLX(M,K)=TMMEAN(K -XFACT*TMRANGE(K)*CONTLIM/3*SUBGRPSIZ(M)
531            IF LCLX(M,K) 0 THEN LCLX(M,K)=0
540            UCLR(M,K)=ULIMR*TMRANGE K )*CONTLIM/
550            LCLR(M,K)=LLIMR*TMRANGE(K)*CONTLIM/3
551            IF LCLR(M,K) 0 THEN LCLR(M,K)=0
               NEXT M
560            RETURN
569 READFL: DISP ,"PROCESSING TYPE",TYPES(K)
570         READ# 1,TYPES K) : POSITION,EXCINC$,FUEL$
571          IF POSITION   0 THEN GOTO READ2FL
572          DISP "TYPE",TYPES(K ,"DOES NOT EXIST"
573          GOTO NEXTK
580 READ2FL: IF FUELIN$    FUEL$ AND FUELIN$    "B" THEN DISP "TYPE",TYPES( ),"N
OT INCLUDED" @ GOTO ADDK
581          READ# 2,POSITION : IND,TOTAL(,)
600         RETURN
607 READALL: DISP ,"PROCESSING TYPE",K
610          READ# 1,K : POSITION,EXCINC$,FUEL$
620          IF POSITION=0 THEN DISP ,"TYPE",K,"DOES NOT EXIST" @ GOTO ADDK
625          IF FUELIN$    FUEL$ AND FUELIN$    "B" THEN DISP "TYPE",K,"NOT INCL
UDED" @ GOTO ADDK
630          GOTO TYPELIM
640 TYPRET: READ# 2,POSITION : IND,TOTAL .)
660          RETURN
670 ACCMEAN: FOR M=1 TO 24
672          IF TOTAL(M+12,2)=0 THEN SUBGRPSIZ(M)=  @ GOTO NEXTM
680          MEAN(M,K)=TOTAL M+12,1)
690          RANGE(M,K)=TOTAL M+12,3)
700        SUBGRPSIZ(M)=TOTAL(M+12,2)
701 NEXTM: NEXT M
710          RETURN
715 ACCUMTOT: FOR M=1 TO 24
720            TOTMEAN M)=TOTMEAN(M)+TOTAL(M+12,1)
725            SUBGRPSIZ(M)=SUBGRPSIZ(M)+TOTAL(M+12,2)
730            IF SUBGRPSIZ(M) = 21 THEN SUBGRPSIZ(M)=20
735 ADDM: NEXT M
738 ACCRNG: FOR N=1 TO 25
740          TOTRNG(N)=TOTRNG(N)+TOTAL(N+11,3)
742          NEXT N
748 RETURN
750 TOTMNRNGE: GOSUB MEANRNGE
751             FOR M=1 TO 24
752             MEAN(M,1)=TOTMEAN(M)
753             RANGE(M,1)=ABS (TOTRNG(M+1)-TOTRNG(M))
754             NEXT M
        K=1
756     GOSUB GRAPHLOGIC
        GOTO FINI
759 GRAPHLOGIC: IF GRAPHTYP   ! THEN FUELTAB$(1)=FUELIN$ ELSE FUELTAB$(B =FUEL$
                GOSUB READSTAT
                RETURN
    ENDTYPE: IF GRAPHTYP=1 THEN GOTO FINI
791          IF GRAPHTYP=2 THEN GOTO TOTMNRNGE
800          L=L+1
810          IF DEPOT(L) 0 THEN GOTO TOTMNRNGE ELSE GOTO SETDE
820 TYPELIM: ON GRAPHTYP GOTO TYP1 ,TYP2 ,TYP3
830 TYP1: IF EXCINC$="NYY" OR EXCINC$="NYN" OR EXCINC$="NNY" OR EXCINC$="NNN" TH
EN DISP ,"TYPE EXCLUDED BY DEFAULT" @ GOTO ADDK ELSE TYPRET
840 TYP2: IF EXCINC$="YNY" OR EXCINC$ "YNN" OR EXCINC$="NNY" OR EXCINC$="NNN" TH
```

```
EN DISP ,"TYPE EXCLUDED BY DEFAULT" @ GOTO ADDF  ELSE TYPRET
850 TYP3: IF EXCINC$="YN" OR EXCINC$="YNN" OR EXCINC$="NYN" OR EXCINC$="NNN" TH
EN DISP ,"TYPE EXCLUDED BY DEFAULT" @ GOTO ADDF  ELSE TYPRET
860 FINI: ASSIGN# 1 TO  *
870       ASSIGN# 2 TO  *
880       ASSIGN# 4 TO  *
881       ASSIGN# 7 TO  *
882       ASSIGN# 5 TO  *
883       DISP ,"PROCESSING [illegible]
884       CHAIN "CNTCHRTPLT"
890 STOP
900 MEANRNGE: ON GRAPHT[illegible] GOSUB CALC1 ,CALC2 ,CALC3
910 RETURN
920 CALC1: READ# 4,POSITION ; IND,TMEAN,TRANGE
930        IF POSITION=0 THEN DISP "MEAN/RANGE NOT LOADED FOR THIS TYPE- PLEASEL
OAD FIRST"
932        TMMEAN(K)=TMEAN
934        TMRANGE(K)=TRANGE
940        RETURN
949 CALC2: READ# 1,100 ; POSITION,EXCINC$,FUEL$
950        READ# 4,POSITION ; IND,TMEAN,TRANGE
960        IF POSITION=0 THEN DISP "MEAN/RANGE NOT LOADED FOR THIS DEPOT-PLEASE
LOAD FIRST"
962        TMMEAN(1)=TMEAN
964        TMRANGE(1)=TRANGE
970        RETURN
980 CALC3: READ# 7,BRANCH ; BRNDESC$,TMEAN,TRANGE
990        IF BRNDESC$="NONE" THEN DISP "MEAN/RANGE OR DESCRIPTION NOT LOADED FO
R THIS BRANCH-PLEASE LOAD FIRST"
992        TMMEAN(1)=TMEAN
994        TMRANGE(1)=TRANGE
1000       RETURN
[illegible]
1020       DISP ,"PROCESSING CONTINUES" @ GOTO ENDTYPE
1030 DIEPET: IF GRAPHTYP=1 THEN FUELIN$="[illegible] @ RETURN
1040 FUEL23: DISP "DO YOU WISH TO PRINT DIESEL OR PETROL(D/P)"
1050         INPUT FUELIN$
1060         IF FUELIN$   "P" AND FUELIN$   "D" THEN GOTO FUEL23
1070         RETURN
1100 REM  ***********     PROGRAM CNTCHART COMPLETE    *******************
 181507
```

```
10 REM    ****************    PROGRAM CUTCHRT     ************************
20 REM    **  THIS PROGRAM IS CALLED BY PROGRAM CONTCHART. IT PLOTS       **
30 REM    **  THE CONTROL CHART FOR THE DATA SET UP IN THAT PROGRAM       **
40 REM    **  IT REQUESTS THE SETTING UP OF THE PAPER IN THE PLOTTER      **
50 REM    **  BEFORE EACH CHART IS PLOTTED.                               **
55 REM    ******************************************************************
60 CLEAR
70 OPTION BASE 1
71 DISP
72 DISP ,"*** CONTROL CHART PLOTS ***"
73 DISP
74 DISP
80  COM RANGE(24,10),MEAN(24,10
90  COM TMMEAN(51),TMRANGE(51)
100 COM UCLX(24,51),LCLX(24,51),UCLR(24,51),LCLR(24,51)
110 COM GRAPHTY ,BRANCH,DEPOT(10),TYPETAB(51),FUELTAB$(51)[1]
120 PLOTTER IS 805
121 DEG
130 L=1
131 M=1
132 I=0
141 ASSIGN# 9 TO  "CONTMAS:D700"
142 READ# 9,1 ; CONTLIM,MONTH,YEAR,SPREAD
143 CONTMONTH=MONTH
150 ASSIGN# 7 TO  "BRANMAS:D700
160 ASSIGN# 8 TO  "DEPMAS:D700"
170 READBRN: READ# 7,BRANCH ; BRNDESC$
180             IF GRAPHTYPE=3 THEN DEPDESC$="ALL" @ GOTO SETUP
190             READ# 8,DEPOT(M) ; DEPDESC$
195 SETUP: LOCATE 22,67,10,90
200         DISP "PLEASE SET UP YOUR PAPER & PRESS C TO CONT OR E TO END"
210             INPUT RESP1$
220             IF RESP1$="E" THEN GOTO FINI
230             IF RESP1$    "C" THEN GOTO SETUP
240 MEANCLC: YMAX=0
250             YMIN=999999
260 SETMINMAX: FOR I=1 TO 24
270              YMIN=MIN (YMIN,MEAN(I,L))
280              YMAX=MAX (YMAX,MEAN(I,L))
290              NEXT I
300              YSUB=YMAX-YMIN
303              IF YSUB 100000 THEN YDIFF=20000
304              IF YSUB   100000 AND YSUB 40000 THEN YDIFF=8000
305              IF YSUB   40000 AND YSUB 2000  THEN YDIFF=4000
310              IF YSUB = 2000 AND YSUB 1000 THEN YDIFF=1000
320              IF YSUB = 1000 AND YSUB 500 THEN YDIFF=100
330              IF YSUB = 500 AND YSUB 100 THEN YDIFF=50
340              IF YSUB = 100 THEN YDIFF=25
350              YMIN=YMIN-YDIFF
360              YMAX=YMAX+YDIFF
370 SCALE YMAX,YMIN,1,24
380 PEN 1 @ LINE TYPE 1
390 FRAME
400 FXD 0
410 AXES YDIFF,1,YMIN,1
420      GOSUB XLABAXES
430 PEN 1
440 LINE TYPE 6
441 MOVE MEAN(1,L),1
450 XCURVE: FOR J=1 TO 24
460          DRAW MEAN(J,L),J
470          NEXT J
```

```
473           LINE TYPE 7 @ PEN 2
479           MOVE UCLX(1,L),1
480 XCONTROL: FOR J=1 TO 24
490           DRAW UCLX(J,L),J
500           NEXT J
501           LINE TYPE 1 @ PEN 2
502           MOVE TMMEAN(L),1
510           FOR J=1 TO 24
520           DRAW TMMEAN(L),J
530           NEXT J
531           LINE TYPE 7 @ PEN 2
532           MOVE LCLX(1,L),1
540           FOR J=1 TO 24
550           DRAW LCLX(J,L),J
560           NEXT J
570 LOCATE 82,127,10,90
580 RNGCALC: YMAX=0
590           YMIN=999999
600 RMINMAX: FOR K=1 TO 24
610           YMIN=MIN (YMIN,RANGE(K,L))
620           YMAX=MAX (YMAX,RANGE(K,L))
630           NEXT K
640           YSUB=YMAX-YMIN
641 IF YSUB 100000 THEN YDIFF=20000
645 IF YSUB = 100000 AND YSUB 50000 THEN YDIFF=10000
650 IF YSUB = 50000 AND YSUB 10000 THEN YDIFF=5000
660 IF YSUB = 10000 AND YSUB 2000 THEN YDIFF=1000
670 IF YSUB = 2000 AND YSUB 500 THEN YDIFF=200
680 IF YSUB = 500 AND YSUB 100 THEN YDIFF=50
690 IF YSUB = 100 THEN YDIFF=25
700 YMAX=YMAX+YDIFF
710 YMIN=YMIN-YDIFF
720 SCALE YMAX,YMIN,1,24
730 PEN 1 @ LINE TYPE 1
740 FRAME
750 FXD 0
760 AXES YDIFF,1,YMIN,1
770       GOSUB RLABAXES
780 PEN 1
790 LINE TYPE 6
800 MOVE RANGE(1,L),1
811 RCURVE: FOR J=1 TO 24
820           DRAW RANGE(J,L),J
830           NEXT J
850 PEN 2
860 LINE TYPE 7
870 MOVE UCLR(1,L),1
880 RCONTROL: FOR J=1 TO 24
890             DRAW UCLR(J,L),J
900             NEXT J
910             LINE TYPE 1
920 MOVE TMRANGE(L),1
930             FOR J=1 TO 24
940             DRAW TMRANGE(L),J
950             NEXT J
960 MOVE LCLR(1,L),1
970 LINE TYPE 7
980             FOR J=1 TO 24
990             DRAW LCLR(J,L),J
1000            NEXT J
1010 SETGU
1020 CSIZE 6
1030 LORG 1 @ LDIR 90
1040 MOVE 3,22
1050 LABALL: LABEL USING FORMA ; "CONTROL CHART",CONTLIM,"SIGMA"
1060 MOVE 12,83
```

```
1070 CSIZE 5
1080 ON CONTMONTH GOSUB MON1 ,MON2 ,MON3 ,MON4 ,MON5 ,MON6 ,MON7 ,MON8 ,MON9 ,MO
N10 ,MON11 ,MON12
1081 LABEL USING FORMC : MONTHDESC$,YEAR
1085 MOVE 12.30
1090 LABEL USING FORMD : "BRANCH:",BRNDESC$
1100 MOVE 17.30
1105 IF GRAPHTYP=3 THEN DEPDESC$="ALL"
1106 GOSUB FUELDESC
1110 LABEL USING FORMB : "DEPOT :",DEPDESC$,FUELDESC$
1120 CSIZE 4
1130 MOVE 19.92
1135 IF GRAPHTYP=3 THEN GOTO VERTAXIS
1140 IF GRAPHTYP=1 THEN GOTO TYP1
1150 LABEL USING "5A" : "TYPES"
1151 IF GRAPHTYP=2 OR GRAPHTYP=3 THEN LABEL USING "K" : "ALL" @ GOTO VERTAXIS
1152 IF TYPETAB(M)=0 THEN GOTO TYPLAB
1153 FOR M=1 TO 15
1154 IF TYPETAB(M)=0 THEN GOTO TYPLAB
1155 NEXT M
1156 LABEL USING "K" : "VARIOUS" @ GOTO VERTAXIS
1160 TYPLAB: FOR M=1 TO 15
1170          IF TYPETAB(M)=0 THEN GOTO VERTAXIS
1180          LABEL USING "K" : TYPETAB(M)
1190          NEXT M
1200 FORMA: IMAGE 14A,"(",D,X,5A,")"
1210 FORMB: IMAGE 8A,K,"(",K,")"
1220 FORMC: IMAGE 4A,DDDD
1225 FORMD: IMAGE 8A,K
1230 VERTAXIS: MOVE 17.0
1240            CSIZE 3
1250            LABEL USING "K/K" : "FUEL","USED"
1260            MOVE 72.25
1270            LABEL USING "6A" : "MONTHS"
1280            MOVE 75.40
1281            LABEL USING "13A" : "MEAN(X) CHART"
1290            MOVE 77.0
1291            LABEL USING "K/K" : "FUEL","RANGE"
1292            MOVE 132.25
1293            LABEL USING "6A" : "MONTHS"
1294            MOVE 135.40
1295            LABEL USING "14A" : "RANGE(R) CHART"
1300 SETUU
1310 IF GRAPHTYP=1 THEN L=L+1 ELSE GOTO FINI
1320 IF L=52 OR TYPETAB(I+1)=0 THEN GOTO FINI
1330 GOTO SETUP
1340 FINI: ASSIGN# 7 TO  *
1350        ASSIGN# 8 TO  *
1360        ASSIGN# 9 TO  *
1370 DISP ,"PROCESSING COMPLETE"
1371 PEN 0
1372 CHAIN  MAINMENU"
1380 STOP
1390 TYP1: I=I+1
1400       LABEL USING "5A/K" : "TYPE",TYPETAB(I)
1410 GOTO VERTAXIS
1420 XLABAXES: LORG 6 @ LDIR 90
1430            FOR M=3 TO 24 STEP 3
1440            MOVE YMIN-.01*ABS (YSUB),M
1450            MONTH=MONTH+3
1460            IF MONTH=13 THEN MONTH=1
1470            IF MONTH=14 THEN MONTH=2
1480            IF MONTH=15 THEN MONTH=3
1490     ON MONTH GOSUB MON1 ,MON2 ,MON3 ,MON4 ,MON5 ,MON6 ,MON7 ,MON8 ,MON9 ,MON
10 ,MON11 ,MON12
1500     LABEL MONTHDESC$
```

```
1510     NEXT M
1520 LORG 6
1530 FOR M=YMIN TO YMAX STEP YDIFF
1540 MOVE M,1
1550 LABEL M
1560 NEXT M
1570 RETURN
1580 RLABAXES: LORG 6 @ LDIR 90
1590           FOR M=3 TO 24 STEP 3
1600           MOVE YMIN-.01*ABS (YSUB),M
1610           MONTH=MONTH+3
1620           IF MONTH=13 THEN MONTH=1
1630           IF MONTH=14 THEN MONTH=2
1640           IF MONTH=15 THEN MONTH=3
1650     ON MONTH GOSUB MON1 ,MON2 ,MON3 ,MON4 ,MON5 ,MON6 ,MON7 ,MON8 ,MON9 ,MON
10 ,MON11 ,MON12
1660     LABEL MONTHDESC$
1670     NEXT M
1680 SETGU
1685 LORG 6
1690 MOVE 137,50
1700 LABEL YEAR-1,"-",YEAR
1710 SETUU
1720 LORG 8
1730 FOR M=YMIN TO YMAX STEP YDIFF
1740 MOVE M,1
1750 LABEL M
1760 NEXT M
1770 RETURN
1780 MON1: MONTHDESC$="JAN"
1790       RETURN
1800 MON2: MONTHDESC$="FEB"
1810       RETURN
1820 MON3: MONTHDESC$="MAR"
1830       RETURN
1840 MON4: MONTHDESC$="APR"
1850       RETURN
1860 MON5: MONTHDESC$='MAY'
1870       RETURN
1880 MON6: MONTHDESC$='JUN"
1890       RETURN
1900 MON7: MONTHDESC$='JUL'
1910       RETURN
1920 MON8: MONTHDESC$= AUG"
1930       RETURN
1940 MON9: MONTHDESC$="SEP"
1950       RETURN
1960 MON10: MONTHDESC$  OCT"
1970        RETURN
1980 MON11: MONTHDESC$='NOV"
1990        RETURN
2000 MON12: MONTHDESC$--"DEC"
2010        RETURN
2020 FUELDESC: IF FUELTAB$ I+1)="P" THEN FUELDESC$="PETROL" @ RETURN
2030           IF FUELTAB$(I+1)="D" THEN FUELDESC$="DIESEL" @ RETURN
2040           FUELDESC$=" "
2050           RETURN
2120 REM  ************    PROGRAM CNTCHRTPLT TERMINATION     *****************
```

```
10   REM  ****************  PROGRAM  CUSUM  *************************
20   REM  **  THIS PROGRAM READS THE FILE FOR A SPECIFIC TYPE  **
30   REM  **  OF GROUP OF VEHICLE TYPES OR ENTIRE DEPOT AND     **
40   REM  **  EVALUATES FOR EACH MONTH. THE CUSUM FIGURE.       **
50   REM  **  FROM THE FILE. THIS IS CALCULATED BY SUBTRACTING **
60   REM  **  A REFERENCE FIGURE FROM EACH MONTHS TOTAL AND     **
70   REM  **  THEN ACCUMULATING THESE TOTALS.                   **
80   REM  **  THESE ARE WRITTEN TO A FILE FOR LATER PLOTTING    **
90   REM  **                                                    **
100  REM  ********************************************************
110 CLEAR
111 DISP
112 DISP ,"***  CUMULATIVE SUM CHARTS ***"
113 DISP
120 OPTION BASE 1
130 DIM TOTAL(36,3),SUMTOT(36)
132 COM TMMEAN(51),TOTCSUM(36,51)
134 COM GRAPHTYP,BRANCH,DEPOT(  ),T,PETAB(  1),FUELTAB$(51)[1]
135 CUSUM=
140 PROCIND=
148 DISP ,"     PROGRAM BEGINS"
149 DISP ," INITIALIZATION SECTION"
150    FOR J=1 TO 36
160        FOR L=1 TO 3
170            TOTAL(J,L)=0
180        NEXT L
181        FOR L=1 TO 51
182            TOTCSUM(J,L)=0
184        NEXT L
185            SUMTOT(J)=0
190    NEXT J
191 ASSIGN# 9 TO  'CONTMAS:D700
192 READ# 9,1 ; CONTLIM,MONTH,YEAR,SPREAD
210 GRAPHDEF: DISP
211             DISP ,"YOU MAY DO ANY OF THE FOLLOWING:"
212             DISP
213             DISP ," 1.PRINT THE CHARTS FOR A SERIES OF TYPES WITHIN A DEPOT"
214             DISP
215             DISP ,"2.PRINT THE TOTAL CHART FOR THE DEPOT'
216             DISP
217             DISP ,"3.PRINT THE TOTAL CHART FOR THE BRANCH"
218             DISP "MAKE YOUR SELECTION"
219             INPUT GRAPHTYP
220             IF GRAPHTYP 3 OR GRAPHTYP 1 THEN GOTO GRAPHDEF
221  BRNCH: DISP "WHICH BRANCH"
230  INPUT BRANCH
231  LET L=1
240  DISP "WHICH DEPOTS TO INCLUDE
241  DISP "(TYPE CODE OR   TO END)"
250  INPUT DEPOT(L)
251  IF DEPOT(L   ) THEN GOTO FINI  ELSE ADDEP
252  NEXTDEP: DI    "MORE  D    T  "
253             DISP "(TYPE CODE OR 0 TO CONT)"
254             INPUT DEPOT(L)
255             IF DEPOT(L)=0 THEN GOTO DEPFIN  ELSE ADDEP
256  ADDEP: IF GRAPHTY    3 THEN GOTO DEPFIN
257         L=L+1
258         GOTO NEXTDEP
260 DEPFIN: GOSUB DIEPET
261         LET K=1
262        IF GRAPHTYP 1 THEN GOTO DISPTYP  ELSE PROCIND=1 @ GOTO SETL
270 DISPTYP: DISP "TYPES TO INCLUDE"
280 TYPEIN: DISP "(TYPE CODE OR   TO END OR 100 FOR DEFAULT)"
```

```
[illegible]
311 NEXTTYPE: [illegible]
[illegible]
330        GOTO [illegible]
331 SETL: LET [illegible]
[illegible]
380 MAINPROC: IF PROCIND$ THEN GOTO [illegible]
390 TYPEPROC: GOSUB [illegible]
[illegible]
```

```
625          IF FUELIN$ [illegible] FUEL$ AND FUELIN$ [illegible] B THEN DISP ,"TYPE [illegible]","NOT INC
LUDED" @ GOTO ADD
6[illegible]0          GOTO TYPELIM
6[illegible]4 TYPRET: READ# [illegible] : IND,TOTAL(,)
660          RETURN
670 ACCCUSUM: FOR M=1 TO 36
672            CUSUM=TOTAL(M,1)-TMEAN+CUSUM
680            TOTCSUM(M,K)=CUSUM
701 NEXTM: NEXT M
710          RETURN
720 ACCUMTOT: FOR M=1 TO [illegible]
725            SUMTOT(M)=SUMTOT(M)+TOTAL(M,1)
729            NEXT M
730            RETURN
735 TOTCUSUM: GOSUB MEANRNGE
740            FOR M=1 TO 36
745            CUSUM=SUMTOT(M)-TMEAN+CUSUM
750            TOTCSUM(M,1)=CUSUM
752            NEXT M
753            TYPETAB(1)=100
754            GOSUB GRAPHLOGIC
756            GOTO FINI
759 GRAPHLOGIC: IF GRAPHTYP [illegible] 1 THEN FUELTAB$(1)=FUELIN$ ELSE FUELTAB$(B)=FUEL
780              RETURN
790 ENDTYPE: IF GRAPHTYP=1 THEN GOTO FINI
791          IF GRAPHTYP=2 THEN GOTO TOTCUSUM
800          L=L+1
810          IF DEPOT(L)=0 THEN GOTO TOTCUSUM ELSE GOTO SETDE
820 TYPELIM: ON GRAPHTYP GOTO TYP1 ,TYP2 ,TYP3
830 TYP1: IF EXCINC$="NYY" OR EXCINC$="NYN" OR EXCINC$="NNY" OR EXCINC$="NNN" TH
EN DISP ,"TYPE EXCLUDED BY DEFAULT" @ GOTO ADD ELSE GOTO TYPRET
840 TYP2: IF EXCINC$="YNY" OR EXCINC$="YNN" OR EXCINC$="NNY" OR EXCINC$="NNN" TH
EN DISP ,"TYPE EXCLUDED BY DEFAULT" @ GOTO ADD ELSE GOTO TYPRET
850 TYP3: IF EXCINC$="YYN" OR EXCINC$="YNN" OR EXCINC$="NYN" OR EXCINC$="NNN" TH
EN DISP ,"TYPE EXCLUDED BY DEFAULT" @ GOTO ADD ELSE GOTO TYPRET
860 FINI: ASSIGN# 1 TO *
870       ASSIGN# 2 TO *
880       ASSIGN# 4 TO *
881       ASSIGN# 7 TO *
882       DISP ,"      PROCESSING COMPLETE"
883       CHAIN "CUSUMPLOT"
890 STOP
900 MEANRANGE: ON GRAPHTYP GOSUB CALC1 ,CALC2 ,CALC3
910 RETURN
920 CALC1: READ# 4,POSITION : IND,TMEAN,TRANGE
930          IF POSITION=0 THEN DISP "MEAN/RANGE NOT LOADED FOR THIS TYPE- PLEASEL
OAD FIRST"
935          TMMEAN(K)=TMEAN
936          TMRANGE(K)=TRANGE
940          RETURN
951 CALC2: READ# 1,100 : POSITION,EXCINC$,FUEL$
952          READ# 4,POSITION : IND,TMEAN,TRANGE
960          IF POSITION=0 THEN DISP "MEAN/RANGE NOT LOADED FOR THIS DEPOT-PLEASE
LOAD FIRST
965          TMMEAN(1)=TMEAN
966          TMRANGE(1)=TRANGE
970          RETURN
980 CALC3: READ# 7,BRANCH : BRNDESC$,TMEAN,TRANGE
990          IF BRNDESC$="NONE" THEN DISP "MEAN/RANGE OR DESCRIPTION NOT LOADED FO
R THIS BRANCH-PLEASE LOAD FIRST"
995          TMMEAN(1)=TMEAN
996          TMRANGE(1)=TRANGE
1000         RETURN
1010 ERR1: DISP ,"THIS BRANCH-DEPOT COMBINATION DOES NOT EXIST"
1020         GOTO BRNCH
1030 DIEPET: IF GRAPHTYP=1 THEN FUELIN$="B" @ RETURN
```

```
1040 FUELLIS: [illegible]
1050          INPUT FUEL[illegible]
1060          [illegible]
1070          RETURN
1100 REM ************   [illegible]   ************************
```

```
596        DRAW [illegible]
597        SETGU
598        LORG 2
599        LABEL USING [illegible]
510        NEXT M
511       PEN 1
512        SETGU
513       LORG 4
520       MOVE [illegible]
530       CSIZE [illegible]
540 LABELL1: LABEL USING FORM1 [illegible]
550        MOVE 120,95
560        CSIZE 5
560        ON CURVENUM GOSUB [illegible]
569 [illegible]
570        LABEL USING FORM2 [illegible]
580        CSIZE 3
590        MOVE 40,85
600        LABEL USING FORM3 [illegible]
610        MOVE 40,80
615        GOSUB FUELDESC
620        LABEL USING FORM3 [illegible]
630        CSIZE 4
640        MOVE 118,75
645        IF GRAPHTYPE=1 THEN GOTO VERTAXIS
650        IF GRAPHTYPE=2 THEN GOTO [illegible]
660        LABEL USING [illegible]
690        LABEL USING "K" [illegible]
710 FORM1: IMAGE 20A
720 FORM2: IMAGE [illegible]
730 FORM3: IMAGE [illegible]
735 FORM4: IMAGE [illegible]
740 VERTAXIS: MOVE [illegible]
750            [illegible]
760            LABEL [illegible]
770            [illegible]
780            LABEL [illegible]
790            [illegible]
800 IF [illegible]
810 IF [illegible]
820 [illegible]
830 [illegible]
840 [illegible]
850 [illegible]
860 [illegible]
865 [illegible]
870 [illegible]
880 [illegible]
890       LABEL [illegible]
900       [illegible]
910 [illegible]
920          FOR [illegible]
930          [illegible]
935          [illegible]
940          IF [illegible]
950          IF [illegible]
960          IF [illegible]
970          [illegible]
 [illegible]
980          LABEL [illegible]
1000         [illegible]
1010         [illegible]
1020         [illegible]
1030         [illegible]
1040         [illegible]
1050         [illegible]
```

```
10[illegible]          [illegible]
10[illegible]          [illegible]
1080          [illegible]
1[illegible]          NEXT [illegible]
1[illegible]          RETURN
1100 MON1: MONTHDESC$="JAN"
1110       RETURN
1120 MON2: MONTHDESC$="FEB"
1130       RETURN
1140 MON3: MONTHDESC$="MAR"
1150       RETURN
1160 MON4: MONTHDESC$="APR"
1170       RETURN
1180 MON5: MONTHDESC$="MAY"
1190       RETURN
1200 MON6: MONTHDESC$="JUN"
1210       RETURN
1220 MON7: MONTHDESC$="JUL"
1230       RETURN
1240 MON8: MONTHDESC$="AUG"
1250       RETURN
1260 MON9: MONTHDESC$="SEP"
1270       RETURN
1280 MON10: MONTHDESC$="OCT"
1290       RETURN
1300 MON11: MONTHDESC$="NOV"
1310       RETURN
1320 MON12: MONTHDESC$="DEC"
1330       RETURN
1340 FUELDESC: IF FUELTYPE$(I+1)="P" THEN FUELDESC$="PETROL" @ RETURN
1350           IF FUELTYPE$(I+1)="D" THEN FUELDESC$="DIESEL" @ RETURN
1360           FUELDESC$=[illegible]
1370           RETURN
1380 REM ****************  PROGRAM [illegible] COMPLETE *****************
```

```
10  REM ******************* PROGRAM: DATAROLL *******************
20  REM ** THIS PROGRAM WILL ROLL BACK ONE MONTH ALL DATA FOR A GIVEN **
30  REM ** DEPOT OR FOR A GIVEN BRANCH. ALL TYPES WITHIN THE DEPOT(OR **
40  REM ** BRANCH) WILL BE ROLLED - THEREFORE ALL DATA FOR THE MONTH  **
50  REM ** MUST ALREADY HAVE BEEN EXTRACTED BEFORE THIS FUNCTION IS   **
60  REM ** CARRIED OUT.                                               **
70  REM ** THE OPTION OF BEING ABLE TO DO THE ROLL BY DEPOT IS IN-    **
80  REM ** CLUDED TO GIVE SOME FLEXIBILITY TO THE SEQUENCE OF         **
90  REM ** OPERATIONS. IT DOES HOWEVER PUT THE RESPONSIBILITY ON THE  **
100 REM ** USER TO KNOW WHICH FILES HAVE BEEN ROLLED AND WHICH NOT.   **
110 REM ****************************************************************
120 CLEAR
130 OPTION BASE 1
135 DIM [illegible]
136 DISP
140 DISP "       *** DATA ROLL FUNCTION ***"
150 DISP
160 SELDISP: DISP "YOU MAY DO 1 OF THE FOLLOWING:"
170          DISP
180          DISP "1. ROLL ALL DATA FOR A GIVEN DEPOT"
190          DISP
200          DISP "2. ROLL ALL DATA FOR A GIVEN BRANCH"
203          DISP
205          DISP "3. END"
210          DISP "PLEASE MAKE YOUR SELECTION"
220          INPUT ROLLTYP
230          IF ROLLTYP<1 OR ROLLTYP>3 THEN GOTO SELDISP
240          ON ROLLTYP GOSUB DEPROLL,BRNROLL,FINI
250          CLEAR
260          GOTO SELDISP
270 DEPROLL: GOSUB BRNCHD
280          GOSUB DEPSND
290          ON ERROR GOTO ERR1
300          ASSIGN# 1 TO [illegible]
310          ASSIGN# 2 TO [illegible]
320          OFF ERROR
330          GOSUB ACTROLL
340          RETURN
350 BRNROLL: GOSUB BRNCHD
360          FOR J=1 TO [illegible]
370          DEPOT$=[illegible]
375          DISP "PROCESSING DEPOT";J
380          ON ERROR GOTO ERR2
390          ASSIGN# 1 TO [illegible]
400          ASSIGN# 2 TO [illegible]
410          OFF ERROR
420          GOSUB ACTROLL
430 NEXTJ: NEXT J
440          RETURN
450 FINI: DISP "PROCESSING COMPLETE"
460       CHAIN "MAINMENU"
470 BRNCHD: DISP "WHICH BRANCH"
480         INPUT BRNCH
490         BRANCH$=VAL$(BRNCH)
500         RETURN
510 DEPSND: DISP "WHICH DEPOT"
520         INPUT DEPOT
530         DEPOT$=VAL$(DEPOT)
540         RETURN
550 ERR1: DISP "THIS BRANCH/DEPOT COMBINATION DOES NOT EXIST"
560       GOTO SELDISP
570 ERR2: DISP "BRANCH ";BRNCH$;" DEPOT ";DEPOT$;" COMBINATION DOES NOT EXIST"
580       GOTO NEXTJ
```

```
590 [illegible]
600         PRINT# [illegible] POSITION [illegible]
610         IF [illegible] THEN DISP [illegible]
615         DISP [illegible]
620         READ# 2,POSITION [illegible]
630           FOR K=1 TO 12
640            TOTAL(K,1)=TOTAL(K,1)+[illegible]
650            TOTAL(K,2)=TOTAL(K,2)+[illegible]
660            TOTAL(K,3)=TOTAL(K,3)+[illegible]
670           NEXT K
680         TOTAL(13,1)=[illegible]
690         TOTAL(13,2)=[illegible]
700         TOTAL(13,3)=[illegible]
710         PRINT# 2,POSITION [illegible] TOTAL(*)
715         DISP [illegible]
720  NEXT J: NEXT I
725  RETURN
730 REM ***************  PROGRAM [illegible] TERMINATE  ***************
```

```
10 REM   ******************  PROGRAM DATAPRNT  *********************
20 REM   **  THIS PROGRAM PRINTS ALL EXISTING CONSUMPTION DATA    **
30 REM   **  FOR A SERIES OF TYPES,THE TOTAL OF A DEPOT OR THE    **
40 REM   **  TOTAL OF A BRANCH.                                   **
50 REM   **********************************************************
60 OPTION BASE 1
62  DIM TOTAL(36,3),DTOTMONCON(36),DTOTANNCON(36),PTOTMONCON(36),PTOTANNCON(36
65 PRINTER IS 701
70 CLEAR
75  DISP
80 DISP ,"*** CONSUMPTION DATA PRINT ***"
90 DISP
100 GRAPHDEF: DISP ,"YOU MAY DO ANY OF THE FOLLOWING"
120 DISP ,"1.PRINT DATA FOR A SERIES OF TYPES"
130 DISP
140 DISP ,"2.PRINT TOTAL DATA FOR A DEPOT"
150 DISP
160 DISP ,"3.PRINT TOTAL DATA FOR A BRANCH"
161 DISP
162 DISP ,"4.END"
163 DISP "MAKE YOUR SELECTION"
170 INPUT GRAPHTYP
180 IF GRAPHTYP 1 OR GRAPHTYP 4 THEN GOTO GRAPHDEF
200 ASSIGN# 9 TO  "CONTMAS:D700"
210 READ# 9,1 ; CNTLIM,CNTMONTH,CNTYEAR,SPREAD
211 MONTH=CNTMONTH
212 YEAR=CNTYEAR
220 PGNO=1
240 ON GRAPHTYP GOTO PRTTYP ,PRTDEP ,PRTBRN ,FINI
250 PRTTYP: GOSUB BRNENQ
260         GOSUB DEPENQ
270         GOSUB TYPENQ
275 ON ERROR GOTO ERR2
279 DISP BRANCH$&DEPOT$&"INX"
280 ASSIGN# 1 TO  BRANCH$&DEPOT$&"INX:D700"
290 ASSIGN# 2 TO  BRANCH$&DEPOT$&"DAT:D700"
295 OFF ERROR
300 FOR J=1 TO 30
310   IF TYPES(J)=0 THEN GOTO FINI
320   READ# 1,TYPES(J) ; POSITION,EXCINC$,FUEL$
330   IF POSITION=0 THEN DISP ,"TYPE",TYPES(J), DOES NOT EXIST" & GOTO NEXTJ
340   READ# 2,POSITION ; IND,TOTAL(,)
350 GOSUB THDR
360 ANNCON=0
370 YEAR=CNTYEAR-3
380 MONTH=CNTMONTH
390        FOR I=1 TO 36
400        MONCON=TOTAL(I,1)
410        ANNCON=TOTAL(I,1)+ANNCON
420        GOSUB DATECALC
430        PRINT USING FORMG ; MONTHDESC$,YEAR,MONCON,ANNCON
440        LNCNT=LNCNT+1
445        IF I=13 OR I=25 THEN ANNCON=0
450        NEXT I
460        PRINT USING "1 ////////// //////////K" ; "","**"
470 NEXTJ: NEXT J
480 FORMG: IMAGE 2X,4A,DDDD,20X,DDDDDD,25X,DDDDDD
490 PRTDEP: GOSUB BRNENQ
500         GOSUB DEPENQ
505         ON ERROR GOTO ERR2
509         DISP BRANCH$&DEPOT$ "INX"
510         ASSIGN# 1 TO  BRANCH$&DEPOT$&"INX:D700"
520         ASSIGN# 2 TO  BRANCH$&DEPOT$&"DAT:D700"
525 OFF ERROR
```

```
526 GOSUB DATINIT
530 GOSUB DATPROC
540 GOSUB DATPRT
550 GOTO FINI
560 DATINIT: FOR K=1 TO 36
570          DTOTMONCON(K)=0
580          DTOTANNCON(K)=0
590          PTOTMONCON(K)=0
600          PTOTANNCON(K)=0
610          NEXT K
615          RETURN
620 DATPROC: FOR J=1 TO 99
630          READ# 1,J ; POSITION,EXCINC$,FUEL$
640          IF POSITION=0 THEN DISP "TYPE ",J,"DOES NOT EXIST" @ GOTO ADDJ
650          READ# 2,POSITION ; IND,TOTAL(,)
670          TANNCON=0
680 IF FUEL$="P" THEN GOTO PACCUM ELSE DACCUM
690 PACCUM: FOR I=1 TO 36
700         PTOTMONCON(I)=PTOTMONCON(I)+TOTAL(I,1)
710         PTOTANNCON(I)=TANNCON+PTOTMONCON(I)
715         IF I=13 OR I=25 THEN PTOTANNCON(I)=PTOTMONCON(I)
720         TANNCON=PTOTANNCON(I)
730         NEXT I
740 GOTO ADDJ
750 DACCUM: FOR I=1 TO 36
760         DTOTMONCON(I)=DTOTMONCON(I)+TOTAL(I,1)
770         DTOTANNCON(I)=TANNCON+DTOTMONCON(I)
780         TANNCON=DTOTANNCON(I)
790         NEXT I
800 ADDJ: NEXT J
810 RETURN
820 DATPRT: GOSUB DHDR
830         YEAR=CNTYEAR-3
840         MONTH=CNTMONTH
850 FOR I=1 TO 36
860 GOSUB DATECALC
870 PRINT USING FORMH : MONTHDESC$,YEAR,DTOTMONCON(I),DTOTANNCON(I),PTOTMONCON(I
),PTOTANNCON(I
880 FORMH: IMAGE 2X,4A,DDDD,14X,DDDDDD,6X,DDDDDD,14X,DDDDDD,6X,DDDDDD
890 LNCNT=LNCNT+1
900 IF LNCNT=66 THEN GOSUB DHDR
910 NEXT I
920 RETURN
930 PRTBRN: GOSUB BRNEND
935         GOSUB DATINIT
940         FOR M=1 TO 5
950         DEPOT$=VAL$ (M)
955         DEPOT=M
960         ON ERROR GOTO ERR1
965         DISP BRANCH$&DEPOT "INX"
970         ASSIGN# 1 TO BRANCH$&DEPOT$& "INX:D700"
980         ASSIGN# 2 TO BRANCH$&DEPOT$& "DAT:D700"
990         OFF ERROR
1000        GOSUB DATPROC
1010 NEXTM: NEXT M
1020 GOSUB DATPRT
1030 GOTO FINI
1039 DHDR: ON CNTMONTH GOSUB MON1 ,MON2 ,MON3 ,MON4 ,MON5 ,MON6 ,MON7 ,MON8 ,MON
9 ,MON10 ,MON11 ,MON12
1040       PRINT USING FORMA : "FUEL CONSUMPTION FIGURES AT",MONTHDESC$,YEAR
1045       ASSIGN#  TO "BRANMAS:D700"
1046       READ#  BRANCH ; BRNDESC$,MEAN,RANGE
1050       PRINT USING FORMB : "BRANCH:",BRNDESC$
1055       ASSIGN# 3 TO "DEPMAS:D700"
1056       READ#  ,DEPOT ; DEPDESC$
1057       PRINT USING FORMB : "DEPOT :",DEPDESC$
```

```
1060       PRINT USING FORMB ; "TYPE: ",TYPES(J)
1061       IF FUEL$="P" THEN FUEL$="PETROL"
1062       IF FUEL$="D" THEN FUEL$="DIESEL"
1063       PRINT USING FORME ; "MONTH/YEAR","FUEL: ",FUEL$,"PAGENO:",PGNO
1075       PRINT USING FORMF ; "THIS MONTH","YEAR TO DATE"
1080       LNCNT=  @ PGNO=PGNO+1
1085       RETURN
1094 DHDR: ON CNTMONTH GOSUB MON1 ,MON2 ,MON3 ,MON4 ,MON5 ,MON6 ,MON7 ,MON8 ,MON
9 ,MON10 ,MON11 ,MON12
1095       PRINT USING FORMA ; "FUE  CONSUMPTION FIGURES AT",MONTHDESC$,YEAR
1100       ASSIGN#   TO  "BRANMAS:D 00"
1101       READ#   BRANCH ; BRNDESC$,MEAN,RANGE
1105       PRINT USING FORMB ; "BRANCH:",BRNDESC$
1110       ASSIGN# 8 TO  "DEPMAS:D700"
1111       READ# 8,DEPOT ; DEPDESC$
1115         IF GRAPHTYP=2 THEN PRTDEPOT$=DEPDESC$ ELSE PRTDEPOT$="ALL"
1125       PRINT USING FORMB ; "DEPOT :",PRTDEPOT$
1130       PRINT USING FORMB ; "TYPES:","ALL"
1135       PRINT USING FORMC ; "MONTH/YEAR","DIESEL","PETROL"
1145       PRINT USING FORMD ; "THIS MONTH","YR TO DATE","THIS MONTH","YR TO DATE"

1155       LNCNT=7 @ PGNO=PGNO+1
1165       RETURN
1175 FORMA: IMAGE 20X,28A,4A,DDDD
1185 FORMB: IMAGE 30X,8A,K
1195 FORMC: IMAGE 2X,10A,15X,6A,30X,6A
1205 FORMD: IMAGE 20X,10A,2X,10A,10X,10A,2X,10A
1215 FORME: IMAGE 2X,10A,15X,6A,K,30X,8A,K
1225 FORMF: IMAGE 30X,10A,20X,12A
1235 DATECALC: MONTH=MONTH+1
1245            IF MONTH=13 THEN MONTH=1 @ YEAR=YEAR+1
1255            ON MONTH GOSUB MON1 ,MON2 ,MON3 ,MON4 ,MON5 ,MON6 ,MON7 ,MON8 ,MO
N9 ,MON10 ,MON11 ,MON12
1265            RETURN
1275 MON1: MONTHDESC$="JAN"
1285        RETURN
1295 MON2: MONTHDESC$="FEB"
1305        RETURN
1315 MON3: MONTHDESC$="MAR"
1325        RETURN
1335 MON4: MONTHDESC$="APR"
1345        RETURN
1355 MON5: MONTHDESC$="MAY"
1365        RETURN
1375 MON6: MONTHDESC$="JUN"
1385        RETURN
1395 MON7: MONTHDESC$="JUL"
1405        RETURN
1415 MON8: MONTHDESC$="AUG"
1425        RETURN
1435 MON9: MONTHDESC$="SEP"
1445        RETURN
1455 MON10: MONTHDESC$="OCT"
1465        RETURN
1475 MON11: MONTHDESC$="NOV"
1485        RETURN
1495 MON12: MONTHDESC$="DEC"
1505        RETURN
1515 BRNENQ: DISP "WHICH BRANCH
1525          INPUT BRANCH
1535          BRANCH$=VAL$ (BRANCH)
1545          RETURN
1555 DEPENQ: IF GRAPHTYP=  THEN RETURN
1565          DISP "WHICH DEPOT"
1575          INPUT DEPOT
1585          DEPOT$=VAL$ (DEPOT)
```

```
1595      RETURN
1605 [illegible]
1615      LET [illegible]
1625      INPUT [illegible]
1630      IF [illegible] THEN [illegible]

1635      GOTO [illegible]
1645 [illegible]
1655 [illegible]
1665      [illegible]
1675      INPUT [illegible]
1685      IF [illegible]
1695      IF [illegible] THEN [illegible]
1697 [illegible]
1705 [illegible]
1710      OFF [illegible]
1715      GOTO [illegible]
1720 [illegible] DOES NOT EXIST"
1723      GOTO [illegible]
1725 [illegible] 1 TO [illegible]
1730      [illegible] 2 TO [illegible]
1740      DISP "[illegible] COMPLETE"
1750      CHAIN "[illegible]"
1755      STOP
1760 REM  ****************** [illegible] TERMINATE  **********************
```

```
1  REM  ************** PROGRAM LISTBRANC [illegible] ***************
10 REM  ***  THIS PROGRAM PRODUCES A LISTING [illegible]            ***
20 REM  ***  [illegible] TABLES TOGETHER WITH THE                   ***
30 REM  ***  DESCRIPTIONS CURRENTLY ASSIGNED TO THEM.               ***
40 REM  **************************************************************
50 OPTION BASE 1
55 PRINTER IS [illegible]
60 ALPHA
70 [illegible]
80          DISP ," ***  LISTING BRANCH [illegible] TABLE  ***"
90          DISP ," [illegible]"
100         DISP
110         DISP ," YOU MAY DO ONE OF THE FOLLOWING:"
120         DISP
130         DISP ,"1) LIST THE TABLE OF BRANCHES"
140         DISP
150         DISP ,"2) LIST THE TABLE OF [illegible]"
160         DISP "MAKE YOUR SELECTION"
161         ASSIGN # 2 TO "CONTROL[illegible]"
162         READ# 2,1 ; [illegible]
163         [illegible]
[illegible]
170         INPUT LISTYP
180         IF LISTYP<1 OR LISTYP>2 THEN GOTO [illegible]
190         ON LISTYP GOSUB LISTBR ,LISTDE
200 FIN1: DISP ,"        PROCESSING COMPLETE"
205       PRINT ,"   *** LISTING COMPLETE ***"
210         [illegible]
220         STOP
230 LISTBR: ASSIGN # 1 TO "[illegible]"
240         [illegible]
250         GOSUB HEADER
260         FOR I=1 TO [illegible]
270         READ# 1,I ; [illegible]
280         PRINT USING FORM3 ; I,[illegible]
290         [illegible]
300         IF [illegible] THEN PRINT USING [illegible] @ GOSUB HEADER
310         NEXT I
320         RETURN
330 LISTDE: ASSIGN # 1 TO "[illegible]"
340         [illegible]
350         GOSUB HEADER
360         FOR I=1 TO [illegible]
370         READ# 1,I ; [illegible]
380         PRINT USING FORM3 ; I,[illegible]
390         [illegible]
400         IF [illegible] THEN PRINT USING [illegible] @ GOSUB HEADER
410         NEXT I
420         RETURN
430 HEADER: IF LISTYP=1 THEN GOSUB TYP1
440         IF LISTYP=2 THEN GOSUB TYP2
450         PRINT USING FORM4 ; [illegible]
460         PRINT USING FORM5 ; [illegible]
470         [illegible]
480         [illegible]
490         RETURN
500 TYP1: PRINT USING FORM1 ; "TABLE OF BRANCHES AND DESCRIPTIONS"
510         PRINT USING FORM1 ; "[illegible]"
520         RETURN
530 TYP2: PRINT USING FORM1 ; "TABLE OF [illegible] AND DESCRIPTIONS"
540         PRINT USING FORM1 ; "[illegible]"
550         RETURN
[illegible]
```

```
580 FORMB: [illegible]
590 FORMC: [illegible]BE [illegible],4[illegible],15[illegible],11[illegible]
600 FORMD: [illegible]E [illegible],DD,10X,[illegible]
610 MON1: MON[illegible]
620       RETURN
630 MON2: MONT[illegible]
640       RETURN
650 MON3: MONTH[illegible]
660       RETURN
670 MON4: [illegible]
680       RETURN
690 MON5: [illegible]
700       RETURN
710 MON6: [illegible]
720       RETURN
730 MON7: [illegible]
740       RETURN
750 MON8: [illegible]
760       RETURN
770 MON9: [illegible]
780       RETURN
790 MON10: [illegible]
800       RETURN
810 MON11: [illegible]
820       RETURN
830 MON12: [illegible]
840       RETURN
850 REM   [illegible]  P-OGR[illegible]M LI[illegible]DEPSRN TERMINATE  [illegible]
```